



**Tomé da Costa Matos Desenvolvimento de um controlador e *software* de
parametrização para caldeiras a *pellets***



**Tomé da Costa Matos Desenvolvimento de um controlador e *software* de
parametrização para caldeiras a *pellets***

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Eletrónica e Telecomunicações, realizada sob a orientação científica do Doutor Paulo Bacelar Reis Pedreiras, Professor Auxiliar do Departamento de Eletrónica Telecomunicações e Informática e do Doutor Valter Filipe Miranda Castelão Silva, Professor Adjunto da Escola Superior de Tecnologia e Gestão de Águeda da Universidade de Aveiro

o júri

Presidente

Prof. Doutor Alexandre Manuel Moutela Nunes da Mota

Professor Associado do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

Prof. Doutor Paulo José Lopes Machado Portugal

Professor Auxiliar da Faculdade de Engenharia da Universidade do Porto

Prof. Doutor Valter Filipe Miranda Castelão Silva

Professor Adjunto da Escola Superior de Tecnologia e Gestão de Águeda da Universidade de Aveiro

Agradecimentos

Agradeço à minha família, especialmente aos meus pais e irmãos pelo incentivo e confiança transmitidos ao longo deste trabalho, mesmo em momentos de algum desnorte.

Ao meu Pai e Tio Lando pela preciosa ajuda prestada em diversas etapas do trabalho.

À Rosa pela compreensão e paciência, além da enorme força que sempre me transmitiu.

Aos meus amigos e colegas que estiveram comigo ao longo destes anos.

Ao Engenheiro Augusto Tavares da ECOAMA pela amabilidade e disponibilidade para a discussão de inúmeros pontos do trabalho.

Aos orientadores, Professor Paulo Pedreiras e Professor Valter Silva pela forma preocupada com que seguiram o desenvolvimento de todo o trabalho, assim como pelo pronto esclarecimento de dúvidas que foram surgindo.

palavras-chave

controlador, *pellets*, recuperador de calor, caldeira, queimador, sistema de controlo, sensores, atuadores.

resumo

O presente trabalho propõe o desenvolvimento de um controlador para aparelhos de aquecimento a *pellets*, como caldeiras, recuperadores de calor e queimadores, em parceria com a empresa de produção e instalação deste tipo de aparelhos, a ECOAMA. O *pellet* tem vindo a ganhar cada vez mais simpatizantes dadas as claras vantagens no seu manuseio, nomeadamente como combustível alternativo ao petróleo para aquecimento de casas e sistemas industriais. O controlador *Fumis ALPHA*, atualmente utilizado, tem provocado alguns constrangimentos tanto a nível funcional como económico para a empresa e a sua substituição por outro que possa ser produzido e customizado pela ECOAMA revela-se um excelente atrativo, além das inúmeras vantagens associadas. A dissertação envolve o projeto e implementação de circuitos de acondicionamento de sinal para sensores e atuadores, envolvendo todo um sistema de controlo.

Várias são as funcionalidades implementadas das quais se destacam a programação horária, a capacidade preditiva, a cooperação entre controladores, etc.

A conceção de uma interface com o utilizador por meio de um *LCD* com funcionalidade de toque permite ao utilizador, seja ele o cliente ou o próprio instalador, uma fácil navegação entre ecrãs, possibilitando a gestão do processo de aquecimento ou a configuração de parâmetros a ele associados. A criação de uma aplicação *PC* vem incrementar as ferramentas ao dispor do fabricante para monitorização, teste e configuração, acrescentando valor ao trabalho.

Verificado o funcionamento do sistema desenvolvido, ficou a certeza de que foi dado um passo firme na criação de um protótipo capaz de poder ser comercializado num futuro próximo.

Keywords

controller, *pellets*, stove, boiler, burner, control system, sensors, actuators

Abstract

This work proposes the development of a controller for pellet heating devices such as boilers, stoves and burners, in partnership with a producer and installer enterprise – the ECOAMA. The pellet has been gaining more and more supporters due to the clear advantages in his handling, particularly as an alternative fuel to oil for heating homes and industrial systems. Fumis ALPHA, the currently controller used has caused some embarrassment to both functional and economic level for the company and its replacement with one that can be produced and customized by ECOAMA proves to be an excellent attraction, in addition to the numerous benefits associated. The dissertation involves the design and implementation of signal conditioning circuits for sensors and actuators, involving a whole control system.

There are several features implemented such as the daily scheduling and predictive capacity, cooperation between controllers, etc.

The conception of a user interface via a touch LCD allows the user, client or installer, easy navigation between screens, enabling management of the heating process or parameters configuration. The creation of a PC application has increased the available tools to the manufacturer for monitoring, testing and configuration, adding value to the work.

After checked and validated the operation of the developed system, it became certain that it was given a firm step in creating a prototype able to be marketed in the near future.

Conteúdo

Capítulo 1	Introdução	1
1.1	Enquadramento	1
1.2	Motivação	1
1.3	Objetivos	2
1.4	Requisitos	2
Capítulo 2	Conceitos e Estado da Arte	5
2.1	<i>Pellets</i>	5
2.1.1	- Processo de fabrico de pellets	7
2.1.2	- Dados estatísticos	8
2.2	Sistemas de aquecimento a <i>pellets</i>	9
2.2.1	- Recuperador de calor	9
2.2.2	- Caldeira	12
2.2.3	- Queimador	13
2.3	Princípio de funcionamento	14
2.4	Regulamentação de emissões	14
2.5	Controladores de aparelhos de aquecimento a biomassa	15
2.5.1	- Fumis ALPHA	19
2.5.2	- Primeira versão do controlador	20
Capítulo 3	Funcionalidades	23
3.1	Armazenamento da data e hora	23
3.2	Interface com <i>LCD + touchscreen</i>	23
3.3	Interface com aplicação <i>PC</i>	24
3.4	Programação do crono termostato	25
3.5	Cooperação de controladores	25
3.6	Predição	25
Capítulo 4	Hardware	27
4.1	Diagrama de blocos	27
4.2	Descrição dos sensores e atuadores	28
4.2.1	- Sensores	28
4.2.2	- Atuadores	30
4.3	<i>Real-Time Clock</i>	32
4.4	Circuitos de acondicionamento de sinal	33
4.4.1	- Sonda NTC	33
4.4.2	- LM335	34
4.4.3	- Termopar	35
4.4.4	- Detetor de passagem por zero	37
4.4.5	- Interface de potência	38

4.5 Microcontrolador	38
4.6 Interface com o utilizador	39
4.6.1 - LCD.....	40
4.6.2 - Módulo VM800C	41
4.7 PCI	42
Capítulo 5 Firmware e Software.....	43
5.1 <i>Firmware</i>	43
5.1.1 - Processo de controlo	43
5.1.2 - Armazenamento e conceptualização dos parâmetros do sistema	47
5.1.3 - Codificação e decodificação da data e hora atual.....	49
5.1.4 - Device drivers	50
5.1.5 - Frequência de amostragem das temperaturas	56
5.1.6 - Criação de primitivas gráficas e “widgets”	57
5.1.7 - Criação de um ecrã	57
5.1.8 - Touch.....	58
5.1.9 - Gestão de ecrãs	58
5.1.10 - Programação da interface com o LCD	61
5.1.11 - Programação do crono termostato.....	62
5.1.12 - Calibração	63
5.1.13 - Predição	63
5.1.14 - Comunicação gateway-controladores	65
5.1.15 - Estrutura da função MAIN	66
5.2 Software para PC	67
5.2.1 - Comunicação com a porta série.....	68
5.2.2 - Estrutura da aplicação.....	71
Capítulo 6 Testes de Validação	77
6.1 Testes aos sensores	77
6.1.1 - NTC	77
6.1.2 - Termopar K.....	79
6.1.3 - LM335.....	80
6.2 Atuadores	81
6.2.1 - Teste em vazio	81
6.2.2 - Teste com cargas individuais	82
6.2.3 - Teste com cargas simultâneas.....	84
6.3 Comunicação entre controladores	86
6.4 Controlador em ambiente real.....	88
Capítulo 7 Conclusões.....	91
7.1 Trabalho Futuro	92

LISTA DE FIGURAS

Figura 2.1 - <i>Pellets</i>	6
Figura 2.2 - Ciclo do CO ₂	6
Figura 2.3 - Secador tambor rotativo.....	7
Figura 2.4 - Mesa de secagem.....	7
Figura 2.5 - Recuperador de calor – versão independente	10
Figura 2.6 - Recuperador de calor integrado com chaminé.....	10
Figura 2.7 - Queimador com alimentação por gravidade.....	11
Figura 2.8 - Queimador com alimentação ascendente	11
Figura 2.9 - Queimador com alimentação horizontal	11
Figura 2.10 - Entrada de ar do próprio compartimento	11
Figura 2.11 - Operação com entrada direta de ar do exterior.....	11
Figura 2.12 - Entrada de ar num canal independente da chaminé.....	11
Figura 2.13 - Caldeira a <i>pellets</i> fabricada em Áustria	13
Figura 2.14 - Queimador ECOAMA	13
Figura 2.15 – Controlador BMC 020X	16
Figura 2.16 - Interface com o utilizador do controlador BMC 020X	16
Figura 2.17 – Controlador ST-91.....	16
Figura 2.18 – Controlador NPBC-V5.....	16
Figura 2.19 – Controlador Fumis ALPHA.....	17
Figura 2.20 - Primeira versão do controlador.....	20
Figura 4.1 – Diagrama de blocos do controlador e suas interfaces.....	27
Figura 4.2 - Sistema de controlo de malha fechada genérico	28
Figura 4.3 - Termístor NTC NS N E 1.5 5/6 E 20 N S E	28
Figura 4.4 - Termopar TC K A 3 4 V 10 I N I S/M	29
Figura 4.5 - LM335	29
Figura 4.6 - Resistência de ignição (HTH Rohrheizer – 79/11/200)	30
Figura 4.7 - Motor de alimentação (BCZ 509).....	31
Figura 4.8 – Ventilador AACO ST25	31
Figura 4.9 – Ventilador AACO VA2 W 50R.....	31
Figura 4.10 - Real-Time Clock	32
Figura 4.11 - Circuito de polarização do Termístor NTC	33
Figura 4.12 - OPAMP em configuração de <i>buffer</i>	34
Figura 4.13 - Circuito de acondicionamento de sinal do LM335.....	35
Figura 4.14 - Circuito de acondicionamento de sinal do termopar	36
Figura 4.15 – Modelo da entrada analógica da ADC do microcontrolador PIC32.....	36
Figura 4.16 - Circuito detetor de passagem por zero da tensão da rede elétrica.....	37
Figura 4.17 – Gráficos da onda retificada (vermelho) e do sinal à saída do detetor (verde).....	37
Figura 4.18 - Interface de potência	38

Figura 4.19 – Microcontrolador PIC32-T795	39
Figura 4.20 - Diagrama de blocos da interface com o utilizador.....	40
Figura 4.21 – LCD NHD-3.5-320240MF-ATXL#-1	40
Figura 4.22 – Módulo VM800C35A.....	41
Figura 5.1 - Estrutura simplificada do código, em C, para uma máquina de estados	44
Figura 5.2 - Diagrama de transição de estados	47
Figura 5.3 - Esquema da rotina de leitura da ADC	51
Figura 5.4 - Fluxograma de sincronização com a tensão da rede elétrica	54
Figura 5.5 - Diagrama de transição entre ecrãs.....	60
Figura 5.6 - Camadas lógicas para interface com o FT800	61
Figura 5.7 - <i>Buffer</i> de comandos gráficos	62
Figura 5.8 - Evolução ideal da temperatura ambiente durante a calibração	64
Figura 5.9 - Esquema de ligação <i>PC-gateway</i> -controladores	65
Figura 5.10 – Campo de dados da trama enviada por cada controlador	65
Figura 5.11 – Campo de dados da mensagem enviada pelo <i>gateway</i>	66
Figura 5.12 - Fluxograma do ciclo infinito da função <i>MAIN</i>	67
Figura 5.13 - Fluxograma da RSI para a receção de caracteres.....	69
Figura 5.14 - Configuração para a leitura de caracteres da porta série	70
Figura 5.15 - Fluxograma da leitura de dados da porta série	71
Figura 5.16 - Fluxograma da escrita de dados na porta série	71
Figura 5.17 - Aplicação inicialmente desconectada.....	72
Figura 5.18 - Troca de mensagens despoletada pelo botão “ <i>Connect</i> ”.....	72
Figura 5.19 - Fluxograma estabelecido pelo botão “ <i>Connect</i> ”	73
Figura 5.20 - Aba para alteração de parâmetros e teste do MA e RI	73
Figura 5.21 - Aba para alteração de parâmetros e teste do VA	74
Figura 5.22 - Ecrã de monitorização do processo de aquecimento.....	75
Figura 5.23 - Troca de mensagens de monitorização entre o <i>PC</i> e o <i>gateway</i>	76
Figura 6.1 – Montagem da experiência em vazio	82
Figura 6.2 – Montagem da experiência com a RI ligada	82
Figura 6.3 - Montagem de teste do barramento <i>CAN</i>	86
Figura 6.4 - Apenas o <i>gateway</i> ligado ao barramento <i>CAN</i>	86
Figura 6.5 - Ambos os microcontroladores ligados ao barramento	87
Figura 6.6 - Remoção do microcontrolador 2.....	87
Figura 6.7 - Alteração dos parâmetros do controlador 1	88
Figura 6.8 - Controlador em teste nas instalações da ECOAMA.....	88
Figura 6.9 - Chama do queimador na fase de “ <i>Ignição</i> ”	90
Figura 6.10 - Chama do queimador no início da fase de “ <i>Queima</i> ”.....	90
Figura 6.11 - Chama do queimador no início da fase de “ <i>Extinção</i> ”.....	90
Figura 6.12 - Câmara de combustão finalizado o processo de aquecimento.....	90
Figura 7.1 – Exemplo de um SC da temperatura interior de uma casa.....	94

Figura 8.1 - Esquema do recuperador de calor Héstia (ECOAMA)	101
Figura 8.2 - Esquema de ligação da RTC ao microcontrolador	102
Figura 8.3 - Desenho da PCI do controlador	102
Figura 8.4 - Processo inativo.....	105
Figura 8.5 - Processo ativo	105
Figura 8.6 - Seleção de nível máximo de queima.....	105
Figura 8.7 - Menu de temperaturas.....	105
Figura 8.8 - Mostrador de temperaturas	106
Figura 8.9 - Seleção de temperatura desejada.....	106
Figura 8.10 - Menu de relógio	106
Figura 8.11 - Atualização da data e hora atuais	106
Figura 8.12 - Programação do crono termostato	106
Figura 8.13 - Predição.....	106
Figura 8.14 - Inserção de palavra-passe	106
Figura 8.15 - Menu de programação.....	106
Figura 8.16 - Menu de parâmetros.....	107
Figura 8.17 - Config. parâmetros do MA	107
Figura 8.18 - Config. parâmetros do VA	107
Figura 8.19 - Config. parâmetros do VE	107
Figura 8.20 - Config. temperaturas	107
Figura 8.21 - Teclado virtual.....	107
Figura 8.22 - Teste aos atuadores	107
Figura 8.23 - Menu de bloqueio do teclado.....	107
Figura 8.24 - Calibração	108
Figura 8.25 - Inserção de nova palavra-passe.....	108
Figura 8.26 - Tramas <i>I2C</i> para transferência de dados	108
Figura 8.27 - Modelo de comunicação <i>SPI</i>	111
Figura 8.28 - Ligação Independente	112
Figura 8.29 - Ligação em cascata	112
Figura 8.30 - Topologia da rede <i>CAN</i>	113
Figura 8.31 - Constituição geral de um nó da rede <i>CAN</i>	114
Figura 8.32 - Controlador <i>CAN</i>	114
Figura 8.33 - Trama <i>CAN Standard</i>	115

LISTA DE TABELAS

Tabela 2.1 - Principais produtores e consumidores de <i>pellets</i> da EU em 2010	8
Tabela 2.2 - Limites de emissões de acordo com o EN 303-5	15
Tabela 2.3 - Características dos controladores do mercado europeu	18
Tabela 3.1 - Ações permitidas a partir do display em função do interveniente	24
Tabela 3.2 - Ações permitidas a partir da aplicação <i>PC</i>	24
Tabela 4.1 - Características dos sensores utilizados.....	30
Tabela 4.2 - Principais características dos atuadores utilizados	32
Tabela 4.3 - Pinos do Conetor J5 do VM800C	41
Tabela 5.1 - Ações em função do estado do processo.....	45
Tabela 5.2 - Estado dos atuadores em função do estado do processo	46
Tabela 5.3 - Mapeamento de memória da RTC.....	49
Tabela 5.4 - Instantes de disparo relativos aos extremos de velocidade dos ventiladores.....	55
Tabela 5.5 - Frequências de amostragem de cada tipo de temperatura	57
Tabela 5.6 - Funcionalidades disponíveis na aplicação <i>PC</i> por tarefa	76
Tabela 6.1 - Corrente e tensão na RI.....	83
Tabela 6.2 - Corrente e tensão no MA.....	83
Tabela 6.3 - Corrente e tensão no VA	83
Tabela 6.4 - Corrente e tensão no VE	83
Tabela 6.5 - Medições com apenas os ventiladores ligados	84
Tabela 6.6 - Medições com os ventiladores e o motor de alimentação ligados	84
Tabela 6.7 - Medições com todos os atuadores ligados.....	85
Tabela 8.1 - Mapeamento dos parâmetros do sistema na memória da RTC	104
Tabela 8.2 - Comandos de atualização de parâmetros	104
Tabela 8.3 - Comandos utilizados no teste dos atuadores	104
Tabela 8.4 - Comandos de agendamento do crono termostato	105
Tabela 8.5 - Comandos solicitação de início e fim de processo.....	105
Tabela 8.6 - Funções de todos os pinos de entrada/saída do LCD.....	116

LISTA DE GRÁFICOS

Gráfico 5.1 - Curva de Tensão-Temperatura da NTC e regressão polinomial de ordem 5.....	51
Gráfico 5.2 - Cálculo da temperatura utilizando apenas um troço.....	52
Gráfico 5.3 - Cálculo da temperatura utilizando dois troços	53
Gráfico 5.4 - Temperatura em função da tensão de saída do MCP602	53
Gráfico 5.5 - Relação entre o Instante de disparo e a tensão aplicada ao VA	55
Gráfico 5.6 - Relação entre o Instante de disparo e a tensão aplicada ao VE	56
Gráfico 6.1 - Medição da temperatura ambiente.....	77
Gráfico 6.2 - Erro associado à medição da temperatura ambiente	78
Gráfico 6.3 - Medição da temperatura da água	78
Gráfico 6.4 - Erro associado à medição da temperatura da água	79
Gráfico 6.5 - Medição da temperatura do ar quente	79
Gráfico 6.6 - Erro associado à medição da temperatura do ar quente	80
Gráfico 6.7 - Medição da temperatura da placa.....	80
Gráfico 6.8 - Erro associado à medição da temperatura da placa.....	81
Gráfico 6.9 - Correntes medidas no teste individual aos atuadores	83
Gráfico 6.10 - Tensões medidas no teste individual aos atuadores	83
Gráfico 6.11 - Corrente total consumida a partir das contribuições individuais vs simultânea...	85
Gráfico 6.12 - Evolução das temperaturas ao longo do processo	89

LISTA DE ACRÓNIMOS

<i>ADC</i>	<i>Analog-to-Digital Converter</i>
<i>API</i>	<i>Application Programming Interface</i>
<i>BCD</i>	<i>Binary-Coded Decimal</i>
<i>BJT</i>	<i>Bipolar Junction Transistor</i>
<i>CAN</i>	<i>Controller Area Network</i>
<i>CI</i>	<i>Circuito Integrado</i>
<i>I2C</i>	<i>Inter Integrated Circuit</i>
<i>LCD</i>	<i>Liquid-Crystal Display</i>
<i>LDR</i>	<i>Light Dependent Resistor</i>
<i>LED</i>	<i>Light-Emitting Diode</i>
<i>MA</i>	<i>Motor de Alimentação</i>
<i>NTC</i>	<i>Negative Temperature Coefficient</i>
<i>OPAMP</i>	<i>Operational Amplifier</i>
<i>PCI</i>	<i>Placa de Circuito Impresso</i>
<i>PIC</i>	<i>Peripheral Interface Controller</i>
<i>PLL</i>	<i>Phased-Locked Loop</i>
<i>QVGA</i>	<i>Quarter Video Graphics Array</i>
<i>RI</i>	<i>Resistência de Ignição</i>
<i>RPM</i>	<i>Rotações por minuto</i>
<i>RSI</i>	<i>Rotina de Serviço à Interrupção</i>
<i>RTC</i>	<i>Real-Time Clock</i>
<i>SC</i>	<i>Sistema de Controlo</i>
<i>SMD</i>	<i>Surface-Mount Technology</i>
<i>SPI</i>	<i>Serial Peripheral Interface</i>
<i>TFT</i>	<i>Thin-Film-Transistor</i>
<i>VA</i>	<i>Ventilador Ambiente</i>
<i>VE</i>	<i>Ventilador de Extração de gases</i>
<i>UART</i>	<i>Universal Asynchronous Receiver/Transmitter</i>
<i>USB</i>	<i>Universal Serial Bus</i>

Capítulo 1 INTRODUÇÃO

1.1 ENQUADRAMENTO

O tema desta dissertação surgiu de um contacto entre a empresa ECOAMA e a Universidade de Aveiro. Esta empresa residente em Águeda, região de Aveiro, dedica-se à produção e instalação de equipamento de aquecimento tendo como combustível a biomassa, mais concretamente o *pellet*. Ao todo são comercializados três tipos de aparelhos: recuperadores de calor utilizados para aquecimento de ar, caldeiras para aquecimento de águas sanitárias e queimadores para pré aquecimento de sistemas como por exemplo fornos. De todos os seus constituintes, apenas o controlador não é fabricado por esta empresa, mas sim importado da Eslovénia, o que pode levar a alguns constrangimentos no que diz respeito à sua customização, em função das preferências dos clientes.

Nasceu daqui uma proposta de dissertação de mestrado, com o objetivo de responder a esta solicitação desenvolvendo um controlador capaz de realizar o controlo de cada um dos aparelhos, possuindo apenas as funcionalidades desejadas e que possa ser livremente configurado pelos engenheiros de desenvolvimento e instalação da ECOAMA.

1.2 MOTIVAÇÃO

O facto de o controlador não ser fabricado na ECOAMA causa alguns transtornos para a empresa, uma vez que sempre que pretendem efetuar uma alteração no *software* (tornar visível na aplicação *PC* um certo parâmetro, por exemplo), por mais simples que ela seja, têm de entrar em contacto com a *ATech Electronics*, a fabricante do controlador usado atualmente. Além do elevado custo associado à alteração solicitada, o tempo total despendido neste processo é muito longo, pese embora os largos anos em que a ECOAMA é sua cliente, o que em nada facilita a tarefa. Em termos de funcionamento, o controlador já foi alvo de algumas queixas por parte de clientes, devido ao facto de se desprogramar aleatoriamente ou se for desligado de um ano para o outro. As saídas são bastante sensíveis a variações de tensão e ficam por vezes sem controlo.

Outra desvantagem deste controlador prende-se com o facto de ser bastante genérico, possuindo inúmeras funcionalidades que não são necessárias entre os aparelhos fabricados.

Para dar resposta a estes inconvenientes, a ECOAMA interessou-se pela possibilidade de desenvolvimento de um controlador, em parceria com a Universidade de Aveiro, que pudesse ser utilizado e programado conforme as suas especificações, tendo em conta sempre o *feedback* do cliente.

Numa primeira abordagem, foi desenvolvida uma dissertação intitulada de “Controlador e *Device Drivers* para um Recuperador de Calor a *Pellets*”, que resultou numa versão funcional de um dispositivo capaz de arrancar e monitorizar o processo de aquecimento num recuperador de calor, não obstante os problemas a ele associados que ficaram por resolver, assim como uma série de funcionalidades que ficaram por acrescentar e o seu custo que deve ser reduzido.

1.3 OBJETIVOS

Tendo em conta o referido acima, foram delineados alguns objetivos para esta dissertação que se encontram enumerados nas próximas linhas:

- Melhoria estética e organização cuidada da placa do controlador relativamente à versão anteriormente desenvolvida, sendo constituído apenas por uma placa;
- Melhoria das interfaces dos sensores e da resolução na medida das temperaturas;
- Otimização da máquina de estados;
- Capacidade de armazenamento permanente de parâmetros do sistema;
- Introdução da funcionalidade de programação do crono termostato;
- Introdução da funcionalidade preditiva;
- Introdução da funcionalidade de comunicação entre controladores;
- Desenvolvimento de uma interface com o utilizador simples e apelativa;
- Desenvolvimento de uma aplicação *PC* para configuração, teste e monitorização do sistema;
- Aproximação do controlador desenvolvido de um protótipo final.

Todos estes objetivos e sua concretização foram ao longo do projeto alvo de discussão e análise com a empresa ECOAMA.

1.4 REQUISITOS

Requisitos correspondem a características que o sistema deverá possuir para atender às necessidades e exigências do utilizador. Assim, após análise do seu propósito, definiram-se os seguintes:

- Medição da temperatura ambiente com uma resolução e precisão de $0,1^{\circ}\text{C}$;
- Medição da temperatura da placa do controlador com uma resolução e precisão de $0,1^{\circ}\text{C}$;
- Medição da temperatura dos gases de escape com uma resolução de $0,1^{\circ}\text{C}$ e uma precisão de 2°C ;
- Cinco níveis de funcionamento dos motores (alimentação e ventiladores);
- Armazenamento permanente da data e hora atual;
- Ações em caso de falha de energia;

Capítulo 2 CONCEITOS E ESTADO DA ARTE

Com o corrente aumento do preço dos combustíveis fósseis torna-se cada vez mais imperativa a necessidade de utilizar formas de produção de energia alternativa no que diz respeito à produção de eletricidade e produção de combustíveis destinados ao aquecimento de casas, automóveis, máquinas industriais, etc. A biomassa é, neste momento, uma possível solução de uma energia renovável, amiga do ambiente permitindo reduzir emissões de gases com efeito de estufa (GEE). A utilização desta fonte energética permite ainda diminuir o risco de incêndios, ao promover um maior aproveitamento dos resíduos florestais, conduzindo a uma melhor limpeza destes espaços, e aumentar a empregabilidade em áreas rurais [1, 2]. Ao reconhecer estes benefícios, a UE, em 2005, começou a estabelecer incentivos para o uso desta fonte de energia, apostando na formação de agricultores, no fornecimento de equipamento para produtores e no investimento de instalações de produção de biocombustíveis [3]. Estimou-se, em 2008, que o uso de energia renovável havia duplicado desde há 35 anos atrás, apesar de apenas corresponder a 13% da energia total consumida no nosso planeta, sendo que 10% correspondia a biomassa [4, 5]. A grande desvantagem deste tipo de energia, quando oriunda da madeira, no seu estado bruto, reside no seu mais baixo poder calorífico comparativamente com os combustíveis fósseis, ao conteúdo de humidade e condições de alimentação. Para contornar esta desvantagem tem-se cada vez mais optado pela utilização de biomassa densificada sob a forma de *pellets*, cuja densidade volúmica e energética é superior quando comparada com a biomassa no seu estado bruto, facilitando o seu transporte e aumentando a eficiência no seu manuseio. Além disso, o seu preço tem vindo a decrescer nos últimos anos [6].

2.1 *PELLETS*

Pellets são um tipo de biocombustível granulado, com forma cilíndrica, que resulta da compressão de diversos tipos de biomassa como cascas e podas de árvore, serradura, maravalha e resíduos resultantes da construção civil e madeiras [6, 7].

A sua forma, tamanho, teor de humidade, densidade volúmica e conteúdo energético uniformes são o que realmente fazem este combustível destacar-se de todos os outros oriundos da biomassa. É esta uniformidade que vai permitir uma alimentação mais fácil e controlada. As suas dimensões variam entre 10mm e 30mm de comprimento e entre 6mm e 10mm de diâmetro. Tipicamente possui um baixo teor de humidade que vai dos 4% aos 8%, comparados com 20% a 60% da biomassa em bruto, o que vai permitir uma maior eficiência na sua combustão. A sua densidade volúmica situa-se na

ordem dos 650 kg.m^{-3} , um valor que pode chegar a ser 4 vezes superior ao da biomassa em bruto, e o seu valor energético é, em média de 4.8 kWh.kg^{-1} , o que permite que uma maior quantidade (massa) de combustível possa ser transportada de cada vez, além do associado conteúdo energético [2, 8, 9].



Figura 2.1 - Pellets [10]

Os benefícios para o ambiente são claros, quando comparados com os combustíveis fósseis. As concentrações de gases poluentes e que contribuem para o efeito de estufa são bastante inferiores, decorrentes da sua produção, transporte e utilização. Consideram-se mesmo neutras as emissões de dióxido de carbono (CO_2), tendo em conta o seu ciclo normal (Figura 2.2) [11].

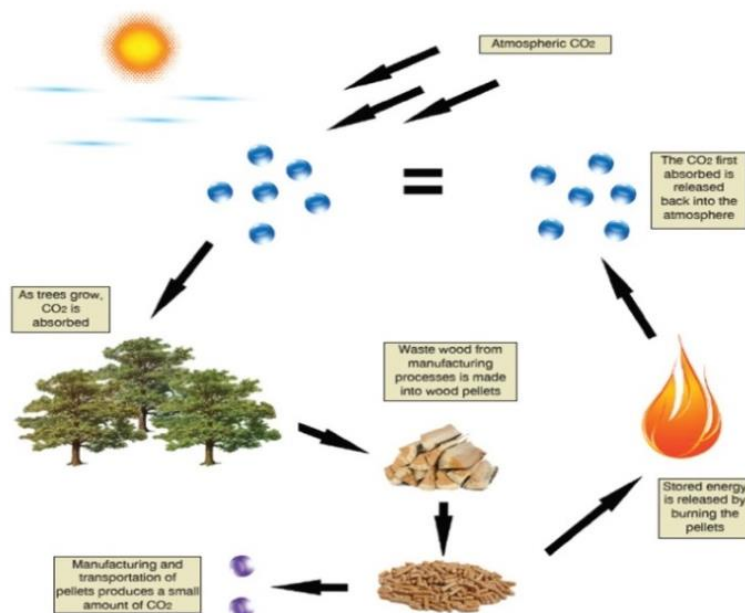


Figura 2.2 - Ciclo do CO_2

2.1.1 - PROCESSO DE FABRICO DE PELLETS

O processo de fabrico de *pellets* envolve várias etapas que devem ser monitorizadas e realizadas sequencialmente. São elas:

1. Receção da matéria-prima – os resíduos florestais encontram-se no seu estado bruto. São pesadas diferentes amostras de material, podendo ser feita uma pré-separação entre conteúdos secos e húmidos. O conteúdo seco deve ser armazenado dentro de um espaço fechado para evitar a formação de poeiras indesejáveis;
2. Secagem – o teor de humidade da matéria-prima é colocado em níveis reduzidos. Esta fase pode ser realizada utilizando dois métodos diferentes. Um deles consiste na utilização de um secador tambor/rotativo (Figura I.3) que funciona a altas temperaturas e é principalmente utilizada para partículas finas, enquanto a mesa secadora (Figura I.4) a temperaturas mais baixas, para partículas de maiores dimensões [12, 13];



Figura 2.3 - Secador tambor rotativo



Figura 2.4 - Mesa de secagem

3. Crivagem – separação de material não resultante da biomassa como pedras, plásticos, pedaços de metal, etc. Esta fase é permutável com a anterior;
4. Trituração ou Moagem – processo de homogeneização da matéria-prima, que reduz as dimensões dos resíduos;
5. Granulação ou Peletização – o *pellet* ganha a sua forma. É utilizado um rolo para comprimir o material contra uma prensa que possui uma matriz aquecida com pequenos orifícios, permitindo a sua formação sob condições de temperatura e pressão elevados;
6. Arrefecimento – fase final da criação do *pellet*, cuja temperatura se encontra na ordem dos 150°C. Através de fluxos de ar frio os *pellets* são arrefecidos e ganham a sua forma final;

7. Verificação – todos os *pellets* que não possuam as dimensões estabelecidas são reintroduzidos no processo.
8. Embalamento – dependendo do cliente final, os *pellets* podem ser embalados em sacos de 12, 15 e 20 kg ou ainda introduzidos num depósito maior de cerca de 1 tonelada. Contudo, os sacos conferem a garantia de maior qualidade do produto na sua entrega, já que ficam menos expostos a agentes abrasivos externos [14-17].

2.1.2 - DADOS ESTATÍSTICOS

Em 2010 foram estimados como os maiores produtores europeus de *pellets* a Alemanha, a Suécia, a Áustria e a Itália com tendência de subida, uma vez que nos cinco anos anteriores os países da União Europeia quadruplicaram a sua produção. Em termos de consumo alguns destes países voltam a aparecer na lista como a Suécia e a Alemanha, a que se juntam a Dinamarca, a Bélgica e a Holanda [18]. Portugal, comparativamente com estes países não é dos que menos produz (430 mil toneladas por ano), contudo o seu consumo é irrisório, exportando praticamente 50% do que produz e armazenando o restante (Tabela 2.1).

País	Produção (kt/ano)	Consumo (kt/ano)	Exportação (kt/ano)
Alemanha	1750	1200	543
Áustria	850	660	285
Bélgica	286	920	50
Dinamarca	180	1600	184
Holanda	120	913	103
Itália	600	850	10
Suécia	1645	2200	69

Tabela 2.1 - Principais produtores e consumidores de *pellets* da EU em 2010

Relativamente à produção de *pellets* existe uma norma estabelecida em 1999 e atualizada em 2010 pela União Europeia através do seu Comité de Normalização para garantir a sua boa qualidade em termos químicos e mecânicos [19]. Essa norma é denominada de EN 14961 e estabelece limites para mais de quinze propriedades do *pellet*, das quais se destacam o diâmetro, comprimento, densidade, o grau de humidade e poeiras, valor calorífico, etc. Esta especificação envolve três grupos diferentes de *pellets* que são classificados quanto à sua origem [20].

2.2 SISTEMAS DE AQUECIMENTO A PELLETS

A maioria dos sistemas de aquecimento a *pellets* de pequena escala encontra-se entre os países mais consumidores deste tipo de combustível, como a Alemanha, a Áustria e a Suécia. Comparativamente com aparelhos convencionais a gás ou gasóleo, vários problemas se levantam, como por exemplo: os requisitos em termos de espaço, a eficiência na conversão de energia, as emissões e a manutenção. Contudo, estas desvantagens podem ser minimizadas e até anuladas utilizando estratégias cuidadas na sua conceção, como por exemplo o dimensionamento do tamanho e potência nominal do aparelho em função das dimensões do espaço envolvente, a sua limpeza regular, a modulação do poder de queima, a otimização da combustão, etc [19].

Os aparelhos a *pellets* utilizados para aquecimento de interiores são genericamente três e serão abordados nas próximas secções. A título de exemplo, o esquema de um recuperador de calor, com várias perspetivas pode ser encontrado na Figura 8.1.

2.2.1 - RECUPERADOR DE CALOR

Este aparelho destina-se ao aquecimento do ar circundante de um espaço fechado, nomeadamente uma sala de uma casa. O calor resultante do aquecimento é transferido através da circulação forçada provocada por um ventilador ambiente (VA) ou por radiação. Existem duas configurações: uma versão independente (Figura 2.5) que necessita de tubagem extra para extração de gases oriundos da combustão e outra com integração em chaminés (Figura 2.6). Ambas apresentam um princípio de funcionamento semelhante, contudo a segunda possui claras vantagens no que toca a requisitos em termos de espaço [21, 22].

A versão independente possui normalmente um reservatório integrado de *pellets* de capacidade limitada, podendo ser utilizado um outro exterior de maiores dimensões. Para informação do utilizador e do próprio controlador acerca do volume de *pellets* presentes, é prática comum a utilização de um sensor de nível [23, 24]



Figura 2.5 - Recuperador de calor – versão independente



Figura 2.6 - Recuperador de calor integrado com chaminé

A alimentação dos *pellets* para a câmara de combustão é fornecida através de um parafuso helicoidal que se encontra ligado a um motor com velocidade de rotação baixa e elevado binário. Em função da direção e sentido de entrada dos *pellets* na câmara de combustão onde se encontra o queimador, três possíveis tipos podem ser considerados. O mais comum designa-se por queimador com alimentação por gravidade (Figura 2.7), em que o parafuso helicoidal conduz os *pellets* desde o reservatório de armazenamento até à câmara de combustão obrigando-os a cair sobre ela, por influência da força gravítica [25]. Outra solução passa pela alimentação do queimador com sentido ascendente (Figura 2.8). Por último, é ainda utilizada a configuração de queimador com alimentação horizontal (Figura 2.9). Dos tipos referidos, a primeira tem a vantagem de assegurar uma separação entre o depósito de armazenamento e a câmara de combustão, além de reduzir o risco de reacendimento e de conseguir um melhor controlo sobre a quantidade de *pellets* introduzida. Em contrapartida, tem a desvantagem deste tipo de alimentação provocar um impacto negativo na combustão, devido à sua constante perturbação. A segunda solução não necessita de um sistema de remoção de cinzas (ao contrário da terceira), uma vez que os *pellets* acabam por fazer esse trabalho guiando-as para fora dessa área. A combustão é bastante estável, contudo, o período de extinção de chama é longo e a probabilidade de reacendimento é elevada [19].

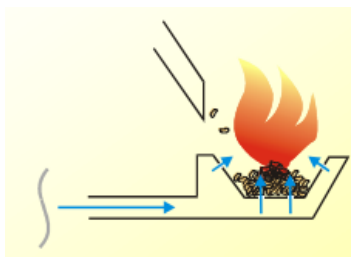


Figura 2.7 - Queimador com alimentação por gravidade

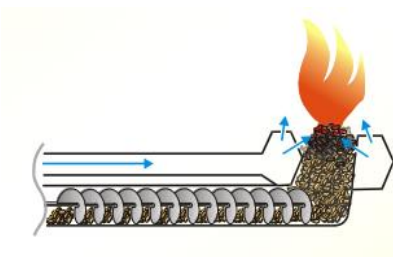


Figura 2.8 - Queimador com alimentação ascendente

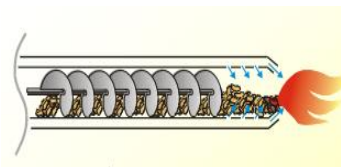


Figura 2.9 - Queimador com alimentação horizontal

A ignição dos *pellets* dá-se utilizando uma resistência elétrica de aquecimento, normalmente cerâmica que atinge a temperatura necessária à ignição dos *pellets* de forma rápida e é, tipicamente, inserida numa ranhura presente na câmara de combustão. Para se obter *feedback* do estado da chama utiliza-se um sensor sensível à luminosidade, tipicamente uma LDR.

O ar que é aproveitado para alimentar a combustão pode entrar na câmara de três formas distintas. Numa delas utiliza-se o próprio ar contido no compartimento que se pretende aquecer (Figura 2.10). A segunda consiste numa entrada independente de ar vindo do exterior (Figura 2.11) e a terceira no aproveitamento do sistema de extração para intercalar um outro canal para entrada de ar (Figura 2.12) [21].

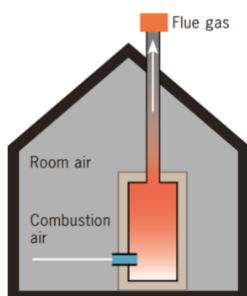


Figura 2.10 - Entrada de ar do próprio compartimento

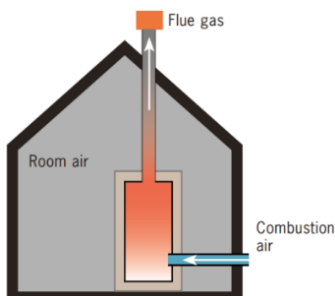


Figura 2.11 - Operação com entrada direta de ar do exterior

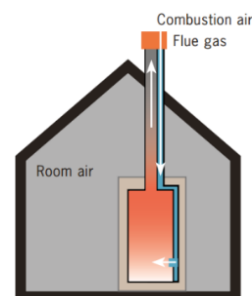


Figura 2.12 - Entrada de ar num canal independente da chaminé

O nível de extração dos gases oriundos da combustão para o ambiente exterior ao espaço fechado, deve ser controlado por um ventilador (VE), em função da sua temperatura e concentração de oxigénio. Para tal é comum utilizar-se um termopar ou Pt1000 e um sensor lambda respetivamente. O efluente produzido pela combustão deve conter uma concentração de oxigénio entre 5% e 8% o que garante uma combustão ótima e uma reduzida emissão de gases poluentes, tais como monóxido de carbono e óxido nítrico. A variação da densidade dos *pellets* que entram na câmara de combustão, pode deteriorar o processo se o fluxo de ar não seguir essa variação [6, 19].

O controlo da temperatura do compartimento onde o recuperador se encontra inserido, deve ter em conta a temperatura desejada pelo utilizador. Se a temperatura ambiente estiver abaixo do valor desejado, o controlador deve automaticamente regular a intensidade de combustão, em função do diferencial de temperatura, para atingir o *setpoint*. Justifica-se, por isso, a instalação um sensor de temperatura ambiente.

2.2.2 - CALDEIRA

O princípio de funcionamento é muito semelhante ao do recuperador de calor, diferenciando-se, principalmente, no seu fim – o aquecimento de águas sanitárias ou outro tipo de líquido. Requisitos de espaço são bastante mais exigentes, uma vez que estes aparelhos possuem dimensões consideráveis necessitando, em grande parte dos casos, de um compartimento anexo para a sua instalação com espaço suficiente para um reservatório integrado de *pellets*, podendo ainda possuir outro para água (Figura 2.13) [26, 27]. Além disso, torna-se imprescindível a utilização de todo um circuito de aquecimento, onde se podem incluir tubos, válvulas, radiadores, serpentina, depósitos, bombas de água, etc. Eventualmente pode, ainda, estar associado a um sistema combinado com energia solar [28].

O ar quente produzido dentro da câmara de combustão encontra o permutador de calor que realiza a transferência de energia dos gases quentes para a água. A limpeza regular deste componente é fulcral para o rendimento do aparelho e sua eficiência energética. Por essa razão, já existem sistemas de limpeza automática [29]. Para colocar a água em circulação, sobre todo o sistema, recorre-se a uma ou mais bombas de água. Os outros atuadores necessários continuam a ser o ventilador de extração de gases, a resistência de ignição e o alimentador de *pellets*, assim como todos os sistemas de limpeza automática de remoção/armazenamento de cinzas e poeiras.

Apesar do controlo efetuado ser sobre a temperatura da água quente que sai da caldeira, o tipo de sensor destinado para a medição da temperatura ambiente pode, muitas vezes ser igualmente utilizado para medição da temperatura da água, como por exemplo um termistor NTC. Todos os restantes sensores abordados no recuperador de calor justificam-se igualmente neste aparelho.



Figura 2.13 - Caldeira a *pellets* fabricada em Áustria

2.2.3 - QUEIMADOR

Trata-se de um dispositivo mais simples, podendo ser integrado tanto em recuperadores de calor como em caldeiras (Figura 2.14). Destina-se essencialmente à produção de calor a altas temperaturas para pré aquecimento de sistemas, como fornos ou secadores industriais. A combustão dá-se de forma ligeiramente diferente, uma vez que a circulação de ar no seu interior se dá por sopro, ao contrário dos aparelhos referidos anteriormente que funcionam por aspiração dos gases de escape. Ao nível de sensores, apenas é necessário o sensor de temperatura para os gases de escape, enquanto os sensores de chama e de nível de *pellets* são opcionais. A resistência de ignição, assim como o ventilador de extração de gases são os principais atuadores necessários deste tipo de aparelho.



Figura 2.14 - Queimador ECOAMA [1]

Como acrescento a estes aparelhos, podem ainda ser introduzidos sistemas de transporte de *pellets*, através de parafuso helicoidal ou sucção aplicando-se a primeira a reservatórios pequenos e a segunda de maiores dimensões. Ao nível da segurança, existem alguns mecanismos contra reacendimentos, em queimadores horizontais ou

ascendentes, como um pequeno sistema de aspersão de água ou ainda mecanismos de separação entre o parafuso alimentador e o queimador.

2.3 PRINCÍPIO DE FUNCIONAMENTO

Como já foi referido, o princípio de funcionamento destes aparelhos é muito semelhante, destacando-se quatro estados principais:

- **Inatividade** - a máquina encontra-se em repouso, não existindo qualquer atividade em execução. O controlador deve encontrar-se em modo de baixo consumo.
- **Ignição** - os *pellets* são transportados para a câmara de combustão e aí são sujeitos a ignição através do calor dissipado por uma resistência de aquecimento;
- **Trabalho/Queima** – a combustão de *pellets* encontra-se em andamento e o calor resultante é transferido para o ar interior de um compartimento no caso do recuperador/queimador ou para a água, se o aparelho for uma caldeira.
- **Extinção** – a chama é extinta de forma gradual para evitar reacendimentos. Deve garantir-se um tempo mínimo nesta fase esperando que a temperatura da câmara de combustão baixe até um certo valor.

É ainda comum acrescentar-se um estado de observação para o caso da temperatura do ar ou da água exceder a desejada e assim evitar todo um reiniciar do processo desde a fase de ignição, quando for necessário voltar a aumentar a temperatura. Desta forma, garante-se uma continuação eficaz do processo e uma melhor performance no que toca à redução de emissões poluentes para a atmosfera [28].

2.4 REGULAMENTAÇÃO DE EMISSÕES

A utilização exaustiva de aparelhos a *pellets* para aquecimento pode resultar na deterioração da qualidade do ar e consequentemente provocar efeitos nefastos para a saúde humana. Assim, o Comité de Normalização da União Europeia criou em 1999 a norma EN 303-5 na qual obriga os produtores deste tipo de equipamento a ter preocupação com a sua conceção no que diz respeito às emissões de certos gases poluentes: o monóxido de carbono (CO), carbono orgânico gasoso (OGC) e poeiras. Nesta norma, dividem-se as caldeiras a *pellets* quanto à sua potência nominal e em três classes diferentes, em função da sua eficiência na transferência de calor para a água (Tabela 2.2):

- Classe 1: 53%-62%
- Classe 2: 63%-72%
- Classe 3: 73%-82%

Nominal heat output kW	Emission limits								
	CO			OGC			dust		
	Mg/m3 at 10% O2 (dry exit flue gas, 00C, 1.013 mbar)								
	class 1	class 2	class 3	class 1	class 2	class 3	class 1	class 2	class 3
>50	15000	5000	3000	1750	200	100	100	180	150
>50 to 150	12500	4500	2500	1250	150	80	200	180	150
>150 to 300	12500	2000	1200	1250	150	80	200	180	150

Tabela 2.2 - Limites de emissões de acordo com o EN 303-5 [30]

As emissões de CO associadas ao funcionamento destes sistemas dependem fortemente das características do aparelho, no tipo de controlo e do número de ignições e extinções efetuadas. Na maioria das unidades de aquecimento a *pellets*, quanto menor for o número de ignições e extinções, menor será a emissão de CO. A modulação da intensidade de combustão pode reduzir o número de ignições e extinções, prolongando o tempo total do processo, enquanto a limpeza automática do queimador e remoção de cinzas asseguram condições estacionárias de combustão com maior eficiência e redução das emissões de CO aumentando o conforto e a aceitação por parte dos clientes [28].

A nível europeu têm sido lançados alguns incentivos e rótulos de qualidade sobre o fabrico de aparelhos de aquecimento a *pellets*, motivando os seus fabricantes a elevarem a qualidade dos seus produtos, garantindo maior visibilidade no mercado atual [19].

2.5 CONTROLADORES DE APARELHOS DE AQUECIMENTO A BIOMASSA

Atualmente são grandes os avanços verificados na indústria dos controladores para aparelhos de aquecimento a *pellets* ou simplesmente a lenha. Consequência disso, são muitas as funcionalidades que estes suportam. Todos eles apresentam uma interface com o utilizador baseada num *display* que pode estar contida num comando de mão ou embutido no próprio aparelho de aquecimento. Em alguns *displays* a funcionalidade de *touch*, sem a presença física de botões já se encontra disponível.

Estes controladores são bastante compactos apesar das muitas interfaces de entrada/saída que já apresentam (ver funcionalidades na Tabela 2.3).

Alguns exemplos de controladores atualmente existentes no mercado são:

- BMC 020X (TEM-Suiça – Figuras 2.15 e 2.16) [31];
- ST-91 (TECH – Polónia – Figura 2.17) [32];
- NPBC-V5 (Naturela Electronics – Bulgária – Figura 2.18) [33];
- Fumis ALPHA (ATech Electronics – Eslovénia – Figura 2.19) [34].

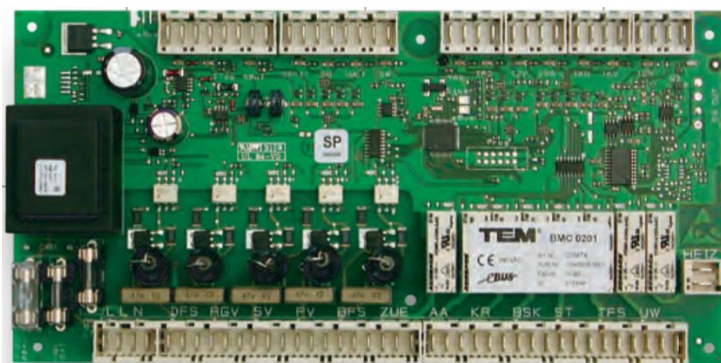


Figura 2.15 – Controlador BMC 020X



Figura 2.16 - Interface com o utilizador do controlador BMC 020X



Figura 2.17 – Controlador ST-91



Figura 2.18 – Controlador NPBC-V5



Figura 2.19 – Controlador *Fumis ALPHA*

Controlador	Aparelho (s)	Modos de Operação	Entradas	Sensores	Atuadores	Saídas	Limpeza	Interfaces
<u>BMC 020X</u>	Caldeira a <i>pellets</i> ou lenha	Manual ou Automático	14	<ul style="list-style-type: none"> • Temperatura ambiente (NTC) • Sensor de temperatura dos gases de escape (PT 1000) • Concentração de Oxigénio (Lambda) • Pressão • Interruptor de porta 	<ul style="list-style-type: none"> • Motor de alimentação • Resistência de ignição • Bomba de água • Ventilador de extração de gases 	12 (230 VAC) 2 (baixa tensão)	Automática	<ul style="list-style-type: none"> • Utilizador <i>Display</i> embutido
<u>ST-91</u>	Caldeira a <i>pellets</i>	Automático	-	<ul style="list-style-type: none"> • Detetor de chama • Interruptor térmico • Temperatura da água quente e do aquecimento central • Temperatura dos gases de escape 	<ul style="list-style-type: none"> • Aquecedor de ignição • Ventilador de extração dos gases de escape • Motor de Alimentação • Duas bombas de água • Duas válvulas de mistura 	-	Manual	<ul style="list-style-type: none"> • Utilizador <i>Display</i> Embutido
<u>NPBC-V5</u>	Caldeira ou Queimador a <i>pellets</i>	Automático	18	<ul style="list-style-type: none"> • Detetor de chama • Temperatura dos gases de escape (Pt100 ou Pt1000) • Temperatura da água na caldeira e no depósito exterior • Temperatura da água no sistema de aquecimento central • Interruptor de porta • Velocidade de rotação do ventilador principal (Sensor de Hall) • Termostato para controlo da temperatura do compartimento 	<ul style="list-style-type: none"> • Alimentador de <i>pellets</i> • Resistência de ignição • Três ventiladores • Quatro bombas de água 	20 (220 ou 110 VAC)	Automática	<ul style="list-style-type: none"> • Utilizador <i>Display</i> com comando • Programador Interface RS232 para conexão com um PC
<u>FUMIS</u> <u>ALPHA</u>	Caldeiras, Queimadores e Recuperadores a <i>pellets</i> ou lenha	Manual ou Automático	12	<ul style="list-style-type: none"> • Temperatura ambiente ou da água (NTC ou Pt100 ou Pt1000) • Temperatura dos gases de escape (dois termopares K) • Nível de pellets • Velocidade de rotação dos ventiladores • Termostato exterior • Interruptor térmico • Pressão • Interruptor de porta • Concentração de oxigénio (Lambda) 	<ul style="list-style-type: none"> • Resistência de ignição • Motor de alimentação • Ventilador de extração de gases de escape • Ventilador ambiente • Bomba de água • Válvula de mistura 	8 (com múltiplas funções)	Automática ou Manual	<ul style="list-style-type: none"> • Utilizador <i>Display</i> com comando ou embutido • Programador Interface RS232 para conexão com um PC

Tabela 2.3 - Características dos controladores do mercado europeu

Analisando a Tabela 2.3 pode concluir-se que os controladores apresentados possuem funcionalidades operacionais muito semelhantes, diferindo no número de interfaces de entrada e saída e nas interfaces com o utilizador e programador.

O último controlador referido na mesma tabela corresponde ao que é utilizado nos equipamentos fabricados pela ECOAMA e terá por isso objeto de uma análise mais extensiva.

2.5.1 - FUMIS ALPHA

Fabricado na Eslovénia pela *ATech Electronics* este dispositivo revela-se como uma avançada ferramenta de controlo de sistemas de aquecimento tendo como combustível a biomassa, seja ela *pellets*, aparas de madeira, lenha, etc.

O seu funcionamento é 100% automático, possibilitando um baixo consumo de combustível e emissões de gases reduzidas. A combustão é otimizada em todos os modos de operação e permite uma redução dos custos associados a este tipo de sistemas. A interface com o utilizador é apelativa e intuitiva, através de um ecrã com vários *displays* de 7 segmentos e botões em posições específicas sensíveis ao toque.

Permite operar com doze configurações diferentes, das quais três se aplicam a recuperadores de calor, sete a caldeiras e as restantes a queimadores. Dependendo da versão do controlador, podem encontrar-se várias disposições de interfaces I/O fornecidas. Podem ainda ser assegurados serviços de monitorização e *fine tuning* de parâmetros através do FUMIS PC-PRO, uma aplicação para PC ou do FUMIS WEB-PRO, uma ferramenta WEB para utilização remota. Complementando estes dois serviços, disponibiliza-se um módulo de monitorização remota via SMS através do FUMIS GSM / GPRS REMOTE CONTROL UNIT G2RCU, que é responsável pelo envio de mensagens de texto para o utilizador sobre o estado do processo [34].

Sendo este um controlador genérico, corre-se o risco de pagar um valor elevado pela sua aquisição, com funcionalidades que podem não ser necessárias. Até ao momento a ECOAMA utiliza este controlador apenas em caldeiras, recuperadores de calor e queimadores exclusivamente a *pellets*, contudo ele pode ser utilizado no mesmo tipo de aparelhos alimentados com outro tipo de combustível. No diz respeito aos sensores, não se estão a utilizar sensores lambda para otimização da combustão, mas apenas sensores de temperatura. Além destes, nenhum sensor de pressão, de deteção de abertura de porta ou de nível de *pellets* se encontra a ser utilizado.

2.5.2 - PRIMEIRA VERSÃO DO CONTROLADOR

No âmbito de um projeto anterior, foi desenvolvido um controlador destinado à operação de um recuperador de calor (Figura 2.20) [35].

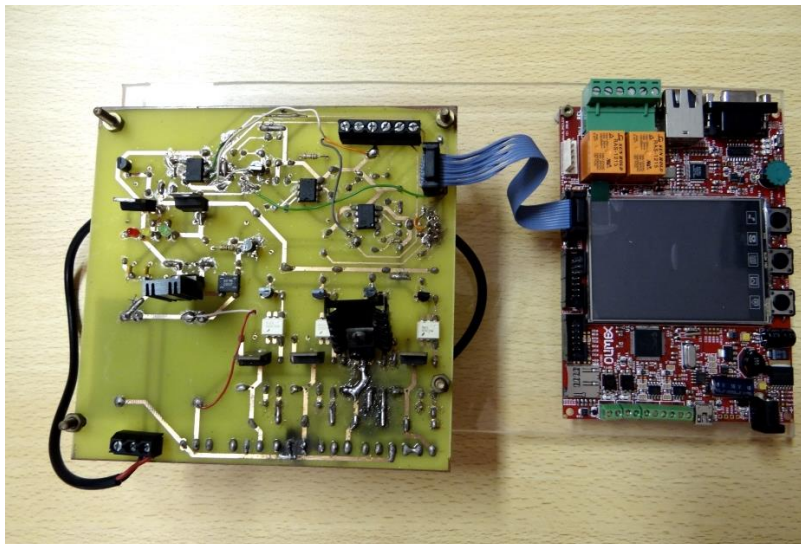


Figura 2.20 - Primeira versão do controlador

Este controlador foi dividido em duas placas. Uma constituída por um microcontrolador (PIC32-MAXI-WEB), periféricos de entrada/saída e ainda um *LCD Touchscreen* incorporado [36]. Na segunda placa encontra-se o *hardware* de acondicionamento de sinal para sensores, a interface de potência para controlo dos atuadores e a fonte de alimentação. Além de possuir dimensões consideráveis (260mm x 140mm), neste controlador não se encontram mecanismos de armazenamento de parâmetros de forma permanente, nem da data e hora. Alguns dos circuitos de acondicionamento possuem pequenos problemas ou precisam de ser melhorados. A resolução na medição de temperaturas é de 1°C e deve ser reduzida, além de ser necessário alargar a gama de medição da temperatura ambiente que se encontra entre 10°C e 30°C, com o objetivo de poder vir, eventualmente, a ser utilizado em países mais frios. O filtro RC utilizado na medição da temperatura dos gases de escape deve ser também revisto. Relativamente à máquina de estados implementada, apesar de ser funcional, não foi elaborada de forma estruturada separando ações de transições entre estados. As funcionalidades apresentadas são reduzidas, sendo apenas possível ligar e desligar o processo de aquecimento (cumprindo a máquina de estados), visualizar temperaturas, selecionar a temperatura desejada, a partir do *LCD*, e comunicar com uma aplicação *PC* para programar alguns parâmetros, ainda que não de forma permanente.

Em termos estéticos este controlador também denota algumas lacunas, uma vez que o desenho da placa não seguiu uma lógica de separação entre pistas e componentes, ficando sujeito mais facilmente a curto-circuitos involuntários.

O preço é inflacionado pela placa de desenvolvimento utilizada, que possui diversos recursos que não são necessários tornando-se, portanto, imperativa a utilização de um microcontrolador mais simples com as características básicas necessárias para a comunicação com outros periféricos, leitura de sinais analógicos, processamento, etc. O *LCD* escolhido deverá ser, assim, introduzido na nova placa de desenvolvimento para o controlador, tendo em conta as características do novo microcontrolador, além de ser necessário realizar toda a programação “de raiz” para a sua interação.

Capítulo 3 FUNCIONALIDADES

No âmbito deste trabalho e tendo em vista a aplicação ao caso específico da operação de um recuperador de calor a *pellets*, foram ponderadas as funcionalidades do controlador a desenvolver. Uma funcionalidade traduz-se num uso especial para que algo seja concebido. Neste âmbito, foram considerados os seguintes grupos de funcionalidades: a interface com o utilizador, a programação horária e a configuração/parametrização. As secções seguintes descrevem as principais funcionalidades implementadas para o controlador de um recuperador de calor. Controlador este que se destina a acionar um conjunto de dispositivos disponíveis, em função das exigências processuais programadas, tendo em conta limites operatórios de modo a oferecer a comodidade e segurança ao utilizador de forma viável.

3.1 ARMAZENAMENTO DA DATA E HORA

Esta funcionalidade é possível utilizando uma *Real-Time Clock*, um circuito integrado propositadamente concebido para armazenamento e contagem do tempo, mesmo em caso de falha de energia. Normalmente é ligada a uma pilha para esse efeito. Através do armazenamento da data e hora no próprio controlador, é permitido ao utilizador a sua consulta e agendamento do processo de aquecimento, assim como o seu acerto.

3.2 INTERFACE COM *LCD* + *TOUCHSCREEN*

Através de um *LCD* com função de toque, é garantida uma interface visual rica e apelativa para o interveniente, seja ele o utilizador normal ou o próprio instalador do sistema. Nesta interface, o utilizador terá um leque de ações que pode manipular consoante as suas preferências. Note-se, contudo, que há uma área reservada a configuração e depuração do controlador, acessível apenas ao instalador ou operador de manutenção através de palavra-passe. A Tabela 3.1 resume as ações que podem ser realizadas por cada tipo de interveniente.

Todos os ecrãs criados foram alvo de discussão e análise em conjunto com a empresa ECOAMA, sendo que a sua versão final é praticamente a idealizada.

Interveniente	Ação
Utilizador/ Instalador	Consulta da data e hora atuais
	Seleção de nível máxima de queima
	Seleção de temperatura desejada
	Visualização de temperaturas
	Alteração da data e hora atuais
	Programação do crono termostato
	Programação do preditor
	Solicitação de início e fim de processo
Instalador	Configuração de parâmetros do sistema
	Teste aos atuadores
	Bloqueio do teclado
	Calibração

Tabela 3.1 - Ações permitidas a partir do *display* em função do interveniente

3.3 INTERFACE COM APLICAÇÃO PC

A aplicação *PC* é um programa de comunicação entre o computador e o controlador. Esta funcionalidade permite realizar algumas das ações proporcionadas pelo *LCD* e acrescenta também outras que não foram incluídas nele por questões ergonómicas. Sempre que ocorrer algum problema com o *LCD*, a aplicação *PC* torna-se importante para verificar se o restante *hardware* se encontra a funcionar como pretendido, facilitando a depuração, além de poder substituir certas ações associadas ao manuseamento da máquina por parte do utilizador (Tabela 3.2). O instalador apenas necessita de ligar um computador ou outro terminal de dados com interface série (RS232/USB), que corra a aplicação, à máquina em questão passando de imediato a comunicar com ela.

Interveniente	Ação
Instalador	Seleção de temperatura desejada
	Visualização de temperaturas
	Programação do crono termostato
	Solicitação de início e fim de processo
	Configuração de parâmetros do sistema
	Teste aos atuadores
	Monitorização do estado do processo
	Monitorização e configuração dos controladores

Tabela 3.2 - Ações permitidas a partir da aplicação *PC*

3.4 PROGRAMAÇÃO DO CRONO TERMOSTATO

Por vezes, surge a necessidade do utilizador pretender agendar previamente o funcionamento do processo de aquecimento para uma certa hora do dia. Este tipo de programação consiste na introdução da hora de início e fim do processo, em horas e minutos, e ainda da temperatura ambiente desejada. O controlador deve, assim, ser capaz de fazer arrancar o processo de aquecimento e finalizá-lo no horário estabelecido pelo utilizador, independentemente se o *setpoint* for atingido.

3.5 COOPERAÇÃO DE CONTROLADORES

Por questões de eficiência e economia, em certos casos torna-se vantajoso instalar um conjunto de caldeiras de aquecimento de águas sanitárias, de menor capacidade, em substituição de uma única caldeira de maior capacidade. Neste cenário, pretende-se que as caldeiras funcionem de forma semiautónoma, entrando cada uma delas em funcionamento apenas quando necessário. Mais concretamente, o número de caldeiras em funcionamento é proporcional ao diferencial entre a temperatura corrente e a temperatura desejada das águas sanitárias. Quando o diferencial é grande, todas as caldeiras deverão entrar em funcionamento, desligando-se à medida que a temperatura for subindo.

Um controlador configurado em modo cooperativo distingue-se dos outros através de um número identificador, da temperatura abaixo da qual entra em funcionamento e do nível de queima. Para configuração destes três parâmetros de cada controlador, utiliza-se um controlador intermediário com funcionalidade de *gateway*. Esta funcionalidade permite que um controlador recolha os parâmetros dos controladores presentes no barramento e se possa ligar a um computador ou terminal de dados para configuração e/ou monitorização.

3.6 PREDIÇÃO

Esta funcionalidade é semelhante à do crono termostato, com a diferença que o utilizador programa a hora a que pretende que o compartimento esteja a uma certa temperatura, ao invés de indicar a que hora este se deve ligar. Assim, autonomamente, o controlador deve ser capaz de calcular a hora a que deve ligar o processo de aquecimento de forma a garantir que a temperatura ambiente atinge a temperatura desejada à hora estipulada. A calibração da máquina ao ambiente envolvente deve ser o primeiro passo a realizar para se poder usufruir desta funcionalidade.

Capítulo 4 HARDWARE

No âmbito deste trabalho, entenda-se *hardware* como um termo coletivo que engloba toda a componente física do controlador, desde fios condutores, resistências, transístores, circuitos integrados, circuitos de acondicionamento, display, motores, fonte de alimentação, etc. O projeto do sistema desenvolvido, iniciou-se com a sua representação através de um diagrama de blocos.

4.1 DIAGRAMA DE BLOCOS

A Figura 4.1 representa o diagrama de blocos projetado para o controlador, assim como sensores e atuadores suportados. Ao longo deste capítulo será dado ênfase a cada um dos blocos identificados, sendo descritos o seu propósito e analisados os seus detalhes.

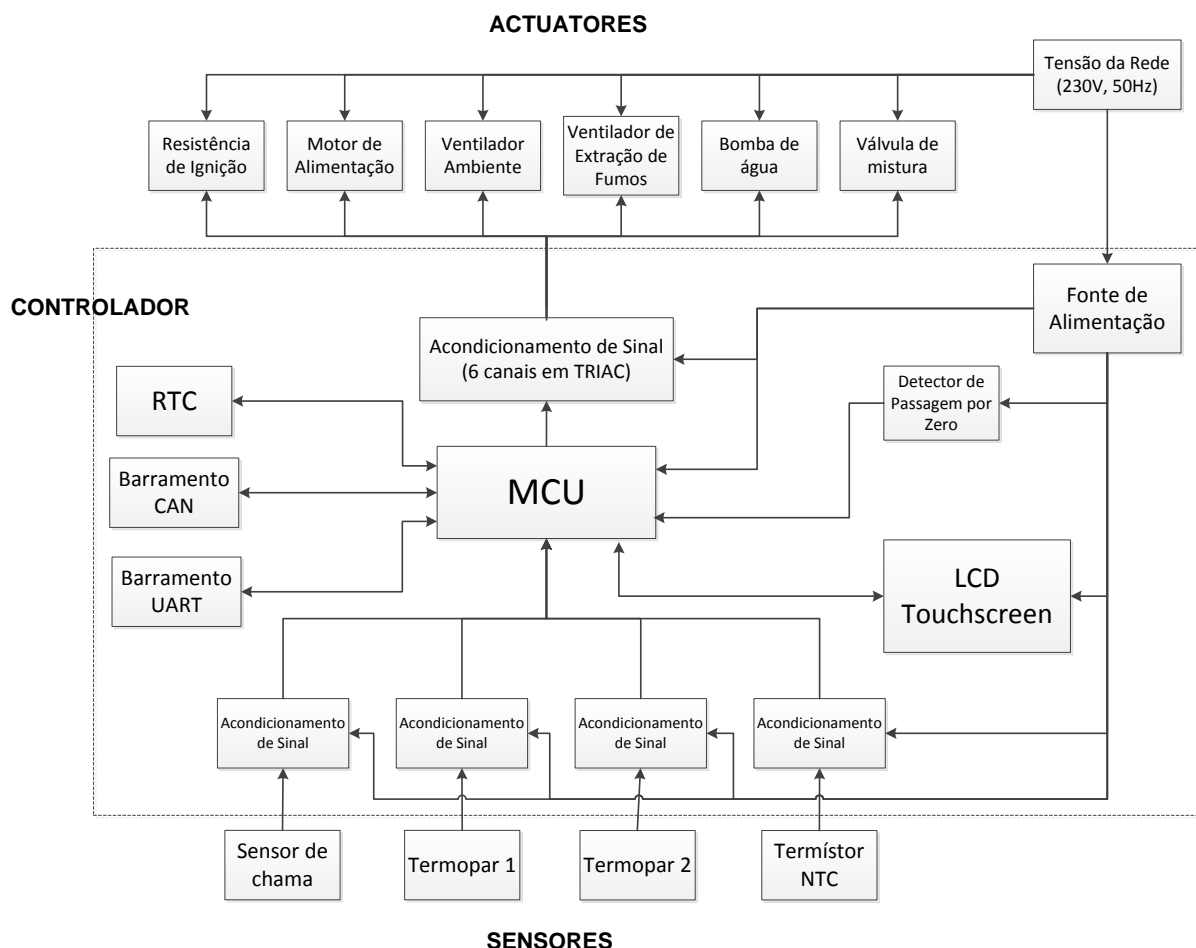


Figura 4.1 – Diagrama de blocos do controlador e sensores e atuadores suportados

4.2 DESCRIÇÃO DOS SENSORES E ATUADORES

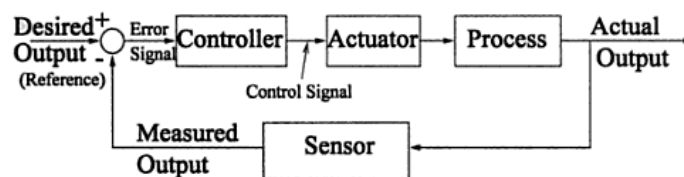


Figura 4.2 - Sistema de controlo de malha fechada genérico [37]

Dois dos principais elementos inerentes a um qualquer sistema de controlo são os sensores, dispositivos que respondem a uma alteração associada a um fenómeno físico, produzindo um sinal proporcional na sua saída e os atuadores, elementos funcionais que estabelecem a ligação entre a parte de processamento de informação de um sistema de controlo e o processo, sendo na maioria das situações responsável pela conversão de energia elétrica em energia mecânica [38, 39].

4.2.1 - SENSORES

Os sensores adotados para este sistema são um termístor *NTC* (Figura 4.3), um LM335 (Figura 4.5) e um termopar tipo K (Figura 4.4), todos sensores de temperatura. O primeiro destina-se, essencialmente, à medição da temperatura do ar ambiente, se estiver inserido num recuperador de calor. Numa caldeira, este tem a função de medir a temperatura das águas sanitárias. Trata-se de um sensor resistivo de coeficiente de temperatura negativo, ou seja, à medida que a temperatura aumenta a resistência aos seus terminais diminui. A sua resposta é altamente não linear, por isso, a expressão para $\theta(R)$ não é trivial, além de que necessita de uma fonte de corrente para a sua polarização. Contudo, apresenta uma boa precisão e sensibilidade. A gama de temperaturas suportada não é extraordinariamente ampla (dos -50°C aos 120°C) mas satisfaz plenamente as necessidades, tanto em ambientes domésticos como industriais [40].



Figura 4.3 - Termístor NTC NS N E 1.5 5/6 E 20 N S E [40]

Já o termopar é um sensor destinado à medição da temperatura dos gases resultantes da combustão de *pellets*, apresentando uma gama de funcionamento superior (0°C aos 350°C), além de ser mais robusto e resistente a pressões mecânicas [40]. Consiste, basicamente, na junção de dois metais de características diferentes que, quando aquecida, gera uma força eletromotriz. A sensibilidade é muito baixa (tipicamente na ordem das dezenas de $\mu\text{V/K}$), pelo que os circuitos de acondicionamento requerem cuidados especiais, pois os sinais na sua saída são normalmente muito pequenos (na ordem de mV). Por ser do tipo K, significa que os dois metais em causa correspondem às seguintes ligas metálicas:

- **Terminal positivo:** Níquel (90%) + Crómio (10%)
- **Terminal negativo:** Níquel (95%) + Alumínio (2%) + Manganésio (2%) + Sílica (1%) [41]



Figura 4.4 - Termopar TC K A 3 4 V 10 I N I S/M

Uma vez que o facto de se ligar o termopar a um aparelho de medida introduz, pelo menos, uma junção “fria” no circuito, torna-se indispensável efetuar a sua compensação.

Para o efeito, utilizou-se mais um sensor de temperatura, neste caso o LM335, um circuito integrado que apresenta uma resposta linear na sua gama de temperaturas de funcionamento: -40°C a 100°C [42]. Este sensor deve medir a temperatura da placa, no ponto onde o termopar é ligado e deve, por isso, ser instalado num bloco isotérmico.



Figura 4.5 - LM335 [43]

De forma sistematizada, esta informação sobre os três sensores em causa, pode ser consultada na Tabela 4.1.

Tipo de Sensor	Função	Gama de Temperaturas	Tipo de resposta	Fabricante	Referência
Termístor NTC	Medição da Temperatura do ar Ambiente/Água	-50°C a 120°C	Não linear	Sacet	NS N E 1.5 5/6 E 20 N S E
Circuito integrado	Medição de Temperatura da Placa	-40°C a 100°C	Linear	Texas Instruments	LM335
Termopar tipo K	Medição da temperatura dos gases de escape	0°C a 350°C	Não linear	Sacet	TC K A 3 4 V 10 I N I S/M

Tabela 4.1 - Características dos sensores utilizados

4.2.2 - ATUADORES

No que diz respeito aos atuadores, destacam-se quatro:

- Resistência de ignição
- Motor de alimentação
- Ventilador ambiente
- Ventilador dos gases de escape

A resistência de ignição (Figura 4.6) é uma fonte de aquecimento de material cerâmico que funciona através da sua ligação à rede elétrica, 120 V ou 220 V, atingindo muito rapidamente a temperatura de trabalho, na ordem dos 1000°C. Constituído por um isolador de Al₂O₃ e por um condutor de platina integrado, é responsável pela queima dos *pellets*. A sua resistência varia de forma linear com a temperatura e apresenta valores típicos de 80 Ω a 20°C e 75 Ω a 0°C [44].

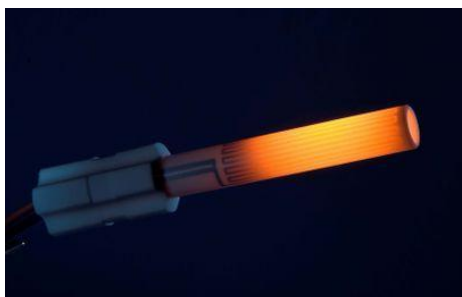


Figura 4.6 - Resistência de ignição (HTH Rohreizer – 79/11/200) [44]

O sistema de alimentação (Figura 4.7) consiste num motor elétrico de indução, ligado a um sem fim, ao qual está acoplado um redutor, e tem como função o transporte de *pellets* desde o depósito onde estes se encontram armazenados até à câmara de combustão. Através de um parafuso transportador helicoidal que roda sobre um eixo inserido no motor, os *pellets* são movimentados e, por ação da força gravítica caem na câmara, ao atingirem a sua extremidade. Este motor possui uma velocidade de rotação fixa e constante na ordem de duas rotações por minuto e uma potência nominal de 55 W.



Figura 4.7 - Motor de alimentação (BCZ 509) [45]

Os ventiladores são também motores elétricos de indução que possuem acoplado diretamente ao veio de um dispositivo de movimentação de ar e são utilizados para realizar tarefas distintas. O ventilador ambiente (Figura 4.8) [46], destina-se a enviar o calor resultante da combustão para o meio ambiente interior, movimentando o ar ambiente através do recuperador de calor. Por sua vez, o ventilador de extração de fumos, ou ventilador do ar primário (Figura 4.9) [47], tem como principal função extrair os gases oriundos da combustão para o meio ambiente exterior, através do recuperador de calor, forçando simultaneamente a entrada de ar no queimador que suporta os *pellets* em combustão.

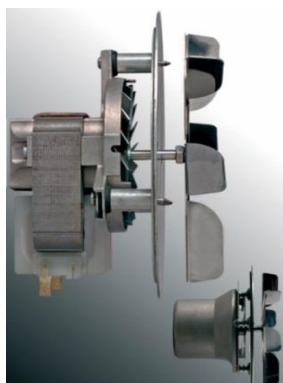


Figura 4.8 – Ventilador AACO ST25



Figura 4.9 – Ventilador AACO VA2 W 50R

A tabela seguinte apresenta as principais características dos atuadores mencionados.

Atuador	Referência	Potência	Temperatura de Trabalho	Fluxo Max. Ar	Velocidade Max. Rotação
Resistência de Ignição	HTH Rohrheizer 79/11/200	175W	980°C	-	-
Motor de Alimentação	BCZ 509	55W	-	-	2 rpm
Ventilador Ambiente	AACO ST25	54W	-25°C a 75°C	122 m³/h	2800 rpm
Ventilador de Escape	AACO VA2 W 50R	55W	-25°C a 150°C	190 m³/h	-

Tabela 4.2 - Principais características dos atuadores utilizados

4.3 REAL-TIME CLOCK

Uma RTC é um relógio que armazena e controla a contagem do tempo de forma autónoma, em tempo real. Este dispositivo adquire-se, normalmente, na forma de um circuito integrado e permite na maioria dos casos a contagem do ano (bissexto ou não), mês, dia do mês, dia da semana, hora (*am* ou *pm*), minuto e segundos. A RTC usada nesta dissertação tem como referência DS1307 (Figura 4.10), possuindo uma memória com capacidade de 64 *bytes* [48]. Os primeiros 8 *bytes* são dedicados à gestão do relógio, sobrando 56 bytes que estão disponíveis para armazenamento de informação de forma permanente.

A interface de comunicação disponibilizada para este circuito integrado é *Inter Integrated Circuit (I2C)*, cujo protocolo se encontra descrito na secção 8.2 dos anexos, assim como o seu circuito de acondicionamento na Figura 8.2.



Figura 4.10 - Real-Time Clock [49]

4.4 CIRCUITOS DE ACONDICIONAMENTO DE SINAL

Os sensores considerados, como visto anteriormente, são específicos para medição de temperatura. Para cada um é atribuída uma gama em que se pretende efetuar essa medição. Estas não devem ultrapassar os limites impostos pelos próprios sensores, para evitar situações que comprometam o normal funcionamento da caldeira, recuperador de calor, ou queimador.

Posteriormente, o circuito deve ser projetado tendo em conta a resposta de cada um dos sensores e os requisitos definidos para o sistema.

4.4.1 - Sonda NTC

O circuito de acondicionamento de sinal utilizado para este sensor foi projetado para se pode medir temperaturas entre os -3°C e os 60°C . Este sensor não linear varia a resistência aos seus terminais em função da temperatura que mede. Projetou-se, assim, um valor de corrente fixo de polarização de $107\mu\text{A}$ que o excitasse através da fonte de corrente da Figura 4.11, com um transistor BJT PNP, duas resistências e um diodo zener.

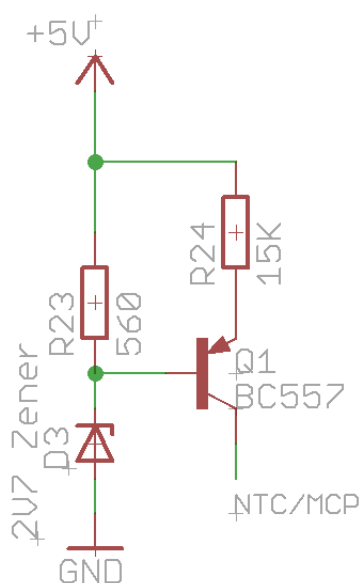


Figura 4.11 - Circuito de polarização do termistor NTC

Utilizou-se, ainda, um *OPAMP rail-to-rail* alimentado a 3.3 V em configuração de *buffer* (Figura 4.12), para evitar que sobretensões na sonda danifiquem o microcontrolador.

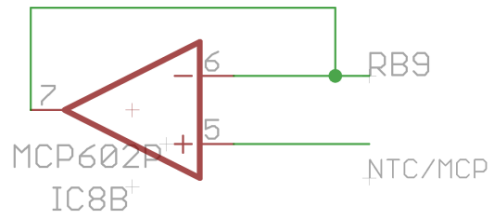


Figura 4.12 - OPAMP em configuração de *buffer*

Recorrendo à tabela presente em [50] pode observar-se que a -3°C a sonda apresenta uma resistência típica aos seus terminais de $31076\ \Omega$, enquanto a 60°C , $3021\ \Omega$. A gama de tensão considerada é, por isso, de $2.988\ \text{V}$. Isto permite obter uma resolução entre os $0,04^{\circ}\text{C}$ para o limite inferior de temperaturas e $0,24^{\circ}\text{C}$ para o limite superior, o que excede ligeiramente o requisito de $0,1^{\circ}\text{C}$, mas para a aplicação em questão não é crítico.

4.4.2 - LM335

A gama de temperaturas a medir por este sensor estabeleceu-se situar entre -20°C e 65°C , sendo que a gama máxima por ele imposta situa-se entre -40°C e 100°C . Sabendo que este sensor produz na sua saída $2.98\ \text{V}$ a 25°C e que a sua resposta varia $10\ \text{mV/K}$, conclui-se que na gama de temperatura considerada se obtém entre $2.53\ \text{V}$ e $3.38\ \text{V}$ na saída do sensor [42]. Além do valor máximo ser superior a $3.3\ \text{V}$, que é a tensão até à qual o microcontrolador consegue ler, esta gama ($0,85\ \text{V}$) é reduzida e não permite que a resolução na leitura da temperatura seja na ordem da décima do grau. Fazendo uma conta simples: $0,85\text{V} / 85^{\circ}\text{C} = 10\text{mV}/^{\circ}\text{C}$ ou $1\text{mV}/(^{\circ}\text{C}/10)$. Como veremos adiante, a resolução da ADC não permite ler intervalos de tensão inferiores a $3,2\ \text{mV}$, logo a resolução da leitura da temperatura, nestas condições, não cumpre os requisitos.

Torna-se, por isso, importante utilizar *hardware* de acondicionamento de sinal para reduzir o *offset* e ampliar a gama de tensão à saída do sensor. Utilizou-se um OPAMP em configuração diferença (Figura 4.13) tendo como tensão de referência um diodo *zener* de $2.4\ \text{V}$, polarizado com uma corrente de $11.2\ \text{mA}$. Admitindo a relação de resistências:

$$\frac{R_{14}}{R_9} = \frac{R_{16}}{R_{10}} \quad (4.1)$$

Obtém-se a expressão geral da tensão à saída deste amplificador:

$$V_{RB0} = (V_{LM335} - V_{Zener}) \frac{R_{16}}{R_{14}} \quad (4.2)$$

Conclui-se que esta tensão depende apenas da diferença entre as tensões do sensor de temperatura e do diodo *zener*.

Importa frisar que “RB0” corresponde ao nome do pino da entrada analógica do microcontrolador que se encontra ligada à saída deste amplificador. Para reduzir o impacto de algumas variações nas tensões de alimentação, utilizou-se um condensador de desacoplamento entre a alimentação positiva e a massa.

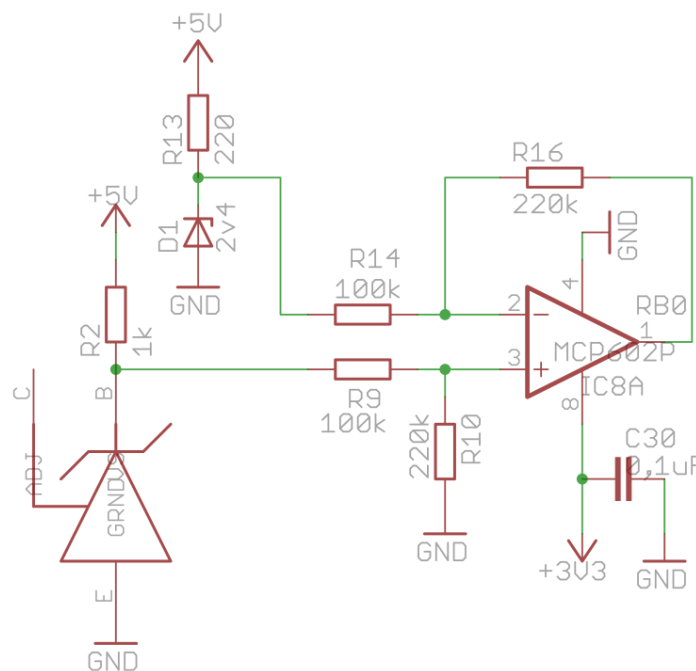


Figura 4.13 - Circuito de acondicionamento de sinal do LM335

4.4.3 - TERMOPAR

Tendo em conta a baixa amplitude do sinal aos terminais deste sensor, na ordem de poucos mV, é necessário amplificá-lo para poder ser adequadamente lido e digitalizado pela ADC. Para tal, utilizou-se o amplificador de instrumentação *rail-to-rail*, AD623N (Figura 4.14), propositadamente concebido para manipular o ganho através de apenas uma resistência, R_G . Tendo em conta que se pretende medir temperaturas entre 0°C e 350°C, projetou-se um ganho de cerca de 180 (expressão 4.3), que resultou na escolha da resistência R_{11} de 560Ω.

$$G = 1 + \frac{10^5}{R_G} \quad (4.3)$$

Utilizaram-se também condensadores de desacoplamento entre as próprias alimentações (3.3 V e massa) para diminuir o efeito de possíveis pequenas oscilações

que possam ocorrer entre as entradas do amplificador e a massa, e ainda apenas entre as duas entradas de forma a impedir que pequenas flutuações afetem o sinal gerado pelo termopar.

Por último, mas não menos importante, importa referir o filtro passa-baixo colocado à saída, constituído pela resistência R_{18} e o condensador C_{10} . O projeto deste filtro teve em conta o modelo da entrada analógica da ADC (Figura 4.15). Ou seja, a resistência R_{18} deve ser pequena o suficiente para permitir que a soma com a resistência R_{IC} e R_{SS} permitam o carregamento total da carga do condensador, C_{HOLD} , para um nível de tensão de até 3.3 V.

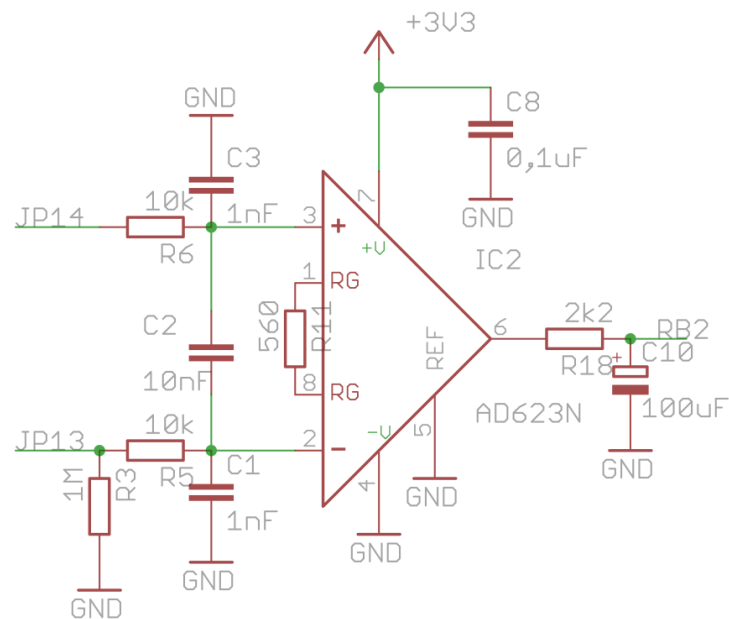


Figura 4.14 - Circuito de acondicionamento de sinal do termopar

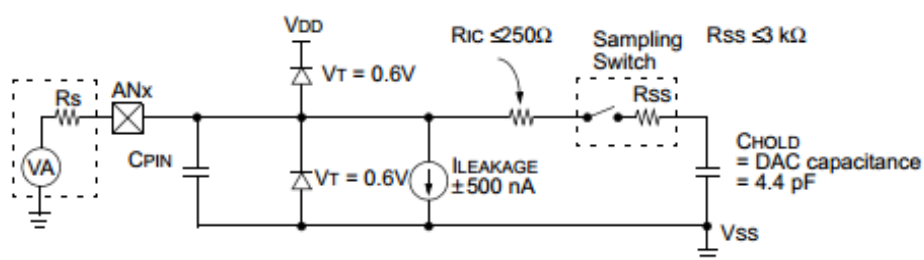


Figura 4.15 – Modelo da entrada analógica da ADC do microcontrolador PIC32

Por outro lado, a frequência de corte do filtro necessita de ser baixa, uma vez que o termopar, principalmente em ambientes industriais, funciona como antena captando sinais de diversas frequências, como por exemplo o da rede doméstica (50 Hz). Escolhida uma frequência de corte baixa e uma resistência que também não pode ser muito alta, o valor do condensador a utilizar terá de ser obrigatoriamente elevado. Escolheu-se uma resistência de $2K2\Omega$ e um condensador de $100\mu F$ cerâmico, selecionando uma frequência de corte de cerca de 0,7 Hz.

4.4.4 - DETETOR DE PASSAGEM POR ZERO

Neste circuito (Figura 4.16) é utilizado o sinal resultante da retificação de onda completa da rede elétrica para “atacar” a base do transístor Q₂. Este vai estar ao corte, sempre que a onda estiver próxima de zero, colocando o seu coletor a 3.3 V. No restante tempo, em que a amplitude da onda faz com que a base do transístor esteja a uma tensão superior ao seu V_{BE}, este fica na zona de saturação. Este processo vai resultar num sinal com picos de 3.3 V de 10 em 10 milissegundos (Figura 4.17) recolhido pelo pino RD11 do microcontrolador, que corresponde a uma entrada de interrupção externa.

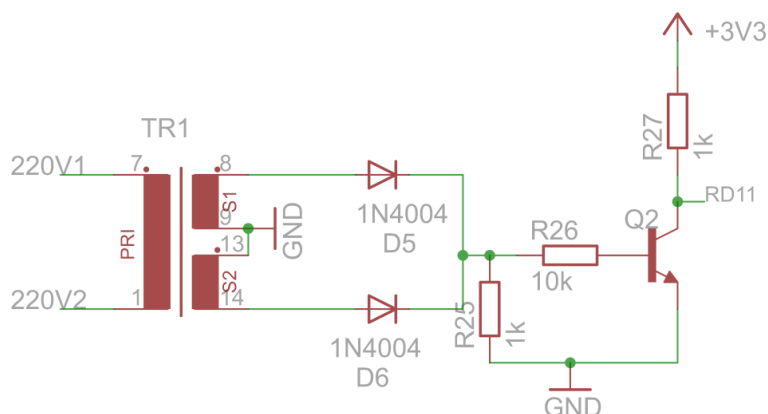


Figura 4.16 - Circuito detetor de passagem por zero da tensão da rede elétrica

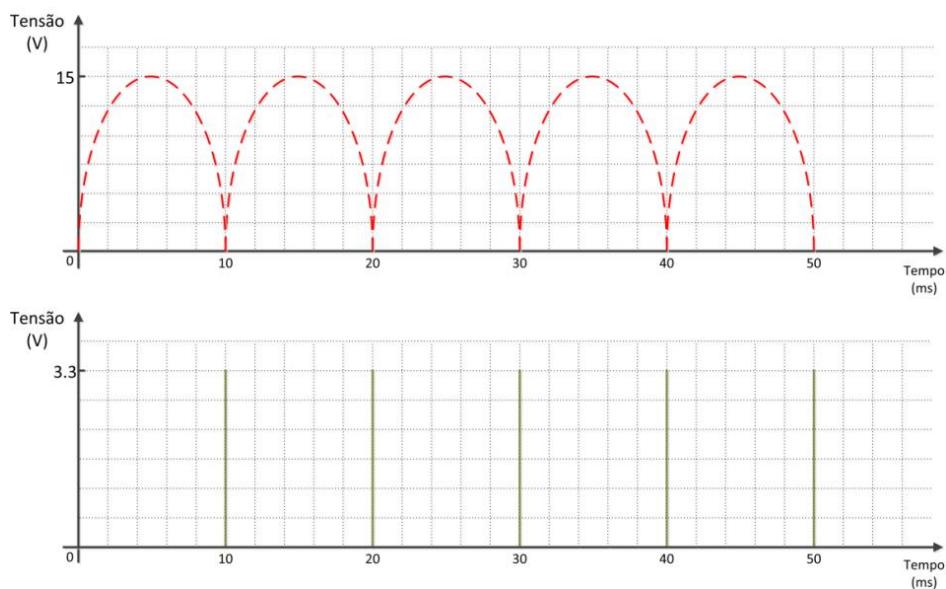


Figura 4.17 – Gráficos da onda retificada (vermelho) e do sinal à saída do detetor (verde)

4.4.5 - INTERFACE DE POTÊNCIA

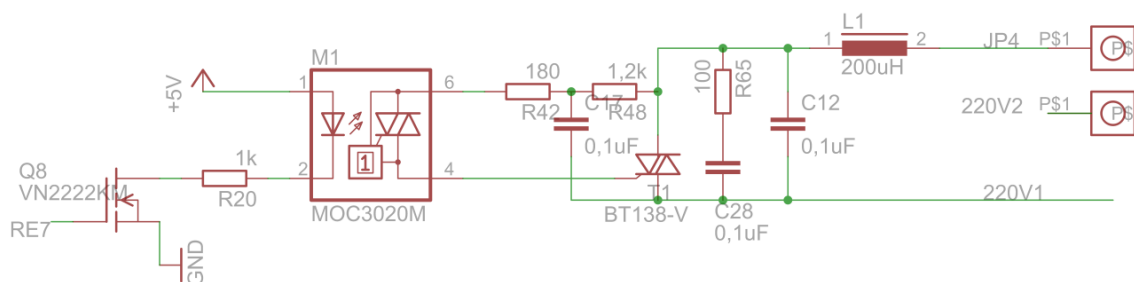


Figura 4.18 - Interface de potência

Para a interface de potência usou-se o circuito *standard*, presente em [51], ao qual foi adicionado um filtro de EMI (Figura 4.18), adequado para cargas indutivas e capacitivas de potência relativamente baixa.

Além de provocar o isolamento entre o sinal elétrico digital de baixa potência e da rede doméstica, o acoplador ótico protege o microcontrolador e todos os restantes circuitos que estão para trás. O *LED* contido no seu interior necessita de uma certa corrente mínima para emitir luz e colocar o foto-*TRIAC* em condução. Esta corrente é garantida através de uma resistência e de um transístor MOSFET, cuja gate se encontra ligada a um pino de saída do microcontrolador. Quando a saída estiver ao nível lógico “0”, o transístor encontrar-se-á cortado e por isso não circula corrente no *LED*, desligando o *TRIAC*. Na situação oposta, com a gate do MOSFET a 3.3 V, superior à sua tensão de *threshold*, V_T (0.6 V), este vai encontrar-se saturado e provocar uma corrente suficiente no *LED* que permite colocar o foto-*TRIAC* em condução. Dada a natureza das cargas, pode justificar-se a utilização de um *snubber* (R_{65} e C_{28}) que suprima transientes de tensão ou corrente no *TRIAC*. Por último, para evitar interferência eletromagnética entre diferentes interfaces de potência, optou-se por utilizar o filtro EMI passa baixo de segunda ordem (C_{12} e L_1).

4.5 MICROCONTROLADOR

A parte central do controlador reside neste dispositivo, que é visto como um computador num *chip*, composto por um microprocessador, memória e periféricos de entrada e saída. O microcontrolador utilizado foi o PIC32-T795 (Figura 4.19), desenvolvido pela OLIMEX para integração em sistemas embutidos.

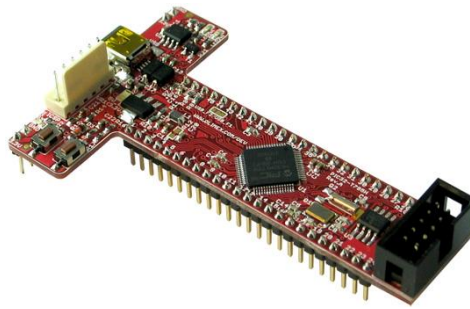


Figura 4.19 – Microcontrolador PIC32-T795

Resumidamente, as características deste microcontrolador são, entre outras, as seguintes:

- Microprocessador *PIC32MX795F512H* de 32 *bits* e frequência máxima de 80 MHz;
- Memória *Flash* de 512 KB;
- Memória *RAM* de 128 KB;
- 3 Módulos *SPI*;
- 4 Módulos *I2C*;
- 2 Módulos *CAN 2.0B*;
- 6 Módulos *UART*;
- 5 *Timers*;
- 16 Canais para *ADC* de 10 *bits*;
- 53 Pinos de entrada/saída;
- Conetor *USB OTG*;
- Conetor *UEXT* que permite a ligação a módulos *RF*, *ZIGBEE*, *GSM*, *GPS*, entre outros;
- Conetor *ICSP* que permite a sua programação através de *PIC-Kit* ou *ICD3* [52]

As principais razões que justificaram a sua escolha prendem-se com a sua simplicidade, possuindo as características necessárias, dimensões reduzidas, o seu baixo custo de aquisição e o facto de a ele estar associado *software* gratuito para a sua programação.

4.6 INTERFACE COM O UTILIZADOR

Em termos esquemáticos, a interface com o utilizador pode ser entendida tal como se encontra representada na Figura 4.20. O *LCD* é, naturalmente, o elemento que disponibiliza a informação de que o utilizador necessita para com ele interagir, seja em termos visuais ou por toque. A placa de desenvolvimento, VM800C, serve de

interveniente entre o microcontrolador e o *LCD*, utilizando as capacidades do processador e controlador gráfico FT800.

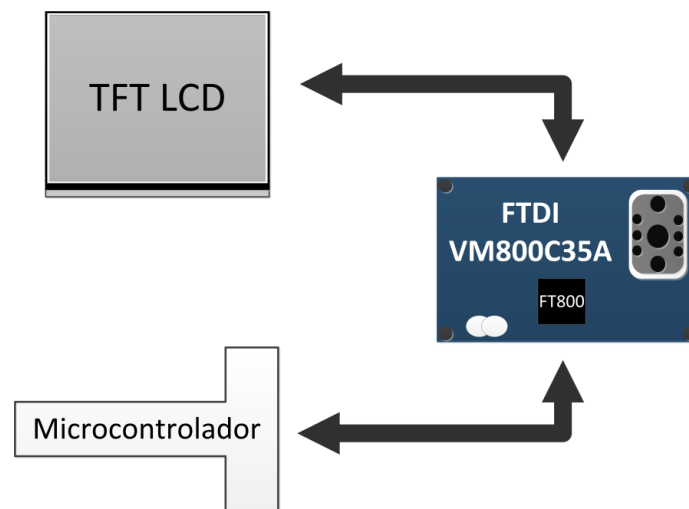


Figura 4.20 - Diagrama de blocos da interface com o utilizador

4.6.1 - *LCD*

O *LCD* utilizado é do tipo *TFT* (Figura 4.21), ou seja, utiliza um tipo de transístores de película fina que melhora a qualidade de imagem relativamente a um *LCD* normal. Possui uma diagonal de ecrã de 3.5 polegadas e uma resolução de 320 x 240 pixéis (QVGA). O *display* é retro iluminado a *LED* na parte traseira às camadas RGB da tela, permitindo um menor consumo de energia que pequenas lâmpadas fluorescentes. De modo a facilitar a sua ligação a dispositivos que controlem o seu funcionamento, este possui um *Flex-Cable* de cinquenta e quatro pinos.



Figura 4.21 – *LCD* NHD-3.5-320240MF-ATXL#-1 [53]

4.6.2 - MÓDULO VM800C

O módulo de desenvolvimento VM800C estabelece uma interface fácil de lidar com aplicações em sistemas embutidos, criando interfaces homem-máquina de alta qualidade. Neste módulo está contido o FT800, da *Future Technology Devices International* (FTDI), um processador gráfico, cujas principais funcionalidades se destacam quanto ao controlo gráfico, de áudio e de toque (resistivo).

Dependendo do tipo e tamanho do *LCD*, existem diferentes versões e configurações para esta placa. O VM800C35A (Figura 4.22) fornece um conector de ligação entre o FT800 e um *TFT LCD display* e 54 pinos para comunicação com o exterior. Na Tabela 8.6 estão discriminados os sinais utilizados em cada um dos cinquenta e quatro pinos do conector.



Figura 4.22 – Módulo VM800C35A

O módulo funciona como *Slave* no que diz respeito à comunicação série *SPI* e necessita, assim, de um microcontrolador adequado para com ele interagir, ou seja, que possua esta interface. É, assim, disponibilizado um conector fêmea de dez pinos, cujos sinais se encontram descritos na Tabela 4.3.

Número do Pino	Sinal	Significado
1	SCK	Entrada de sinal de relógio
2	MOSI	Entrada de dados do Slave
3	MISO	Saída de dados do Slave
4	CS#	Entrada de Seleção do Slave
5	INT#	Saída de Interrupção
6	PD#	Sinal de ativação da retroiluminação do <i>LCD</i>
7	5V	Alimentação 1
8	3.3V	Alimentação 2
9	GND	Massa
10	GND	Massa

Tabela 4.3 - Pinos do conector J5 do VM800C

Este módulo apresenta ainda outras características, tais como:

- *Driver LED* para retroiluminação;
- Saída para áudio mono;
- Fonte de alimentação flexível – *USB Micro-B* ou conector de dois pinos para 3.3V ou 5V externos;
- Cristal interno de 12MHz com *PLL* que pode fornecer até 36MHz ou 48MHz de frequência de relógio.

Os motivos que pesaram na escolha deste módulo têm a ver com as suas elevadas capacidades gráficas, resolução, facilidade de manipulação, o sensor de toque integrado e o seu custo moderado de aquisição.

4.7 PCI

A placa de circuito impresso (PCI) desenvolvida (Figura 8.3) possui dimensões de 188 mm por 189 mm e foi desenhada segundo uma estratégia de simplicidade, segurança e para facilitar o seu *debug* através de aparelhos de medida. Para isso foram seguidas algumas regras:

- Preocupação de colocar todas as pistas debaixo da placa (1 camada) e os componentes em cima, apesar de em alguns casos ter sido necessário a utilização de *jumpers*;
- A interface de potência encontrar-se claramente separada da alimentação e de pistas de pequeno sinal;
- Utilização de pistas com largura considerável, assim como *pads* bem revestidos de área cobreada.

A área respeitante ao retângulo de linha azul traduz o espaço ocupado (por cima dos componentes) pelo *LCD* e pela estrutura que o suporta.

Esta placa possui 6 saídas em *TRIAC* o que permite controlar até 6 atuadores diferentes, como por exemplo resistências de ignição, alimentadores de *pellets*, extratores de fumos, ventiladores, bombas de água, etc. Relativamente aos sensores estão colocadas 4 entradas, sendo duas delas para termopar, outra para Termístor *NTC* e ainda uma para sensor de chama, que não foi desenvolvida nesta dissertação. Esta versatilidade permite que a PCI concebida possa ser utilizada tanto em recuperadores de calor, queimadores ou caldeiras.

Capítulo 5 FIRMWARE E SOFTWARE

Neste capítulo será abordado, em primeiro lugar o *Firmware*, o qual interage com o *hardware* de forma autónoma, cumprindo os requisitos estabelecidos. Posteriormente, será dado foco ao *Software*, através de uma aplicação *PC*, a qual tem o objetivo de ser utilizada pelo fabricante do sistema para configuração de parâmetros, análise do seu funcionamento e manutenção.

5.1 FIRMWARE

O *firmware* foi desenvolvido com o auxílio das ferramentas MPLAB X IDE e XC32, disponibilizadas pela *Microchip*. Este tem por objetivo realizar variadas ações, tais como ler dados dos sensores, interagir com dispositivos com diferentes interfaces de comunicação, tomar decisões autonomamente, controlar atuadores e processar dados adequadamente.

5.1.1 - PROCESSO DE CONTROLO

O controlo utilizado para este sistema é em malha fechada, uma vez que o sinal de saída, medido pelo sensor de temperatura ambiente é utilizado para se combinar com o sinal de referência, ou seja, com o sinal que representa o objetivo a atingir (a temperatura desejada), resultando no sinal de erro que excita o controlador. Em função do sinal de erro, que pode ser maior ou menor consoante a diferença entre as temperaturas de referência e ambiente, o controlador aplica um sinal proporcional no sistema com o objetivo de conduzir o sinal de saída até ao valor pretendido [54]. O controlo proporcional está dividido em cinco níveis de diferentes amplitudes no que diz respeito ao grau de queima. O nível máximo é aplicado quando um grande sinal de erro é gerado, enquanto o nível mínimo corresponde à situação em o mesmo é pequeno.

Para projetar o processo de controlo recorreu-se a uma máquina de estados. Esta pode ser entendida como um modelo abstrato de um sistema ou comportamento e está na maioria dos casos associada a representações de sistemas de controlo que monitorizam aplicações. Como o nome indica, esta representação envolve um conjunto de estados e sua interligação, por meio de sinais. Estes sinais estão relacionados com o conhecimento que se tem da aplicação e permitem a ocorrência de transições entre esses estados. Em cada instante, a máquina pode somente estar num deles, o denominado estado atual [55, 56].

O sistema de controlo abordado nesta dissertação possui cinco estados, sendo eles:

- Inativo
- Ignição
- Queima
- Extinção
- Observação

Com o objetivo de tornar a escrita do código de implementação de uma máquina de estados mais simples, em linguagem C, assim como uma leitura mais legível da mesma, existem algumas regras a ter-se em conta e que se encontram esquematicamente representadas na Figura 5.1:

- Enumeração dos estados do sistema através de MACROS
- Declaração e inicialização da variável que armazena o estado atual
- Utilização de um “*switch*” para seleção do estado atual
- Separação entre ações associadas ao estado de condições de transição

```
#define S0      0
#define S1      1
...
#define SN      N
void StateMachine() {
    int state = S0; // or whatever is the initial state.
    while (1) {
        switch (state) {
            S0:
                // Insert S0's actions here & Insert transitions Ti leaving S0:
                if( T0's condition is true ) {state = T0's next state; /*actions*/ }
                if( T1's condition is true ) {state = T1's next state; /*actions*/ }
                ...
                if( Tn's condition is true ) {state = Tn's next state; /*actions*/ }
                break;
            S1:
                // Insert S1's actions here
                // Insert transitions Ti leaving S1
                break;
            ...
            SN:
                // Insert SN's actions here
                // Insert transitions Ti leaving SN
                break;
        }
    }
}
```

Figura 5.1 - Estrutura simplificada do código, em C, para uma máquina de estados [57]

Para se perceber melhor o seu funcionamento, na Tabela 5.1 está contida uma descrição individual das ações tomadas pelo controlador em cada um dos estados.

Estado	Ações
Inativo	- Manter os atuadores desligados, a não ser que estejam a ser executados testes de funcionamento dos mesmos.
Ignição	<ul style="list-style-type: none"> - É iniciado um contador de tempo (em minutos) - O alimentador de <i>pellets</i> encontra-se inicialmente ligado durante um certo intervalo de tempo. Ao fim desse tempo, caso não tenha ocorrido transição, começa a funcionar segundo um <i>duty-cycle</i> baixo; - A resistência de ignição encontra-se sempre ligada, assim como o ventilador do ar primário (gases) com uma certa velocidade constante.
Queima	<ul style="list-style-type: none"> - A resistência de ignição encontra-se desligada; - Tanto o alimentador de <i>pellets</i> como os ventiladores funcionam segundo um nível que é calculado pela diferença entre as temperaturas desejada e ambiente ou entre a temperatura máxima permitida para os gases e o seu valor instantâneo. A primeira modulação inicia-se quando for atingido um erro inferior a 5°C, a segunda quando o erro for inferior a 30°C. Ambas funcionam de forma conjunta; - Caso a temperatura dos gases baixe até valores inferiores à temperatura mínima de funcionamento da ventoinha ambiente, esta é desligada, para evitar que o ar arrefeça o compartimento.
Extinção	<ul style="list-style-type: none"> - É iniciado um contador de tempo (em segundos). - Garante que o alimentador de <i>pellets</i> e a resistência de ignição se encontram desligados, enquanto os ventiladores funcionam com velocidade de rotação máxima para extrair o calor de dentro da câmara de combustão o mais depressa possível; - Caso a temperatura dos gases baixe até valores inferiores à temperatura mínima de funcionamento da ventoinha ambiente, esta é desligada, para evitar que o ar arrefeça o compartimento.
Observação	- Não toma nenhuma ação, a não ser verificar as temperaturas ambiente e dos gases de escape.

Tabela 5.1 - Ações em função do estado do processo

De forma mais específica, a Tabela 5.2 indica o estado (ligado ou desligado) de cada atuador em função da fase do processo. À primeira vista pode parecer que as fases “Inativa” e de “Observação” são iguais, mas de facto isso não é verdade. Na primeira, que corresponde ao estado inicial, o processo de aquecimento ainda não se iniciou, ou

já terminou, enquanto em “Observação”, o processo encontra-se ativo, nomeadamente com a temperatura ambiente desejada atingida e ultrapassada em pelo menos 1°C.

<div> <div>ATUADOR</div> <div>ESTADO</div> </div>	Resistência de Ignição	Motor de Alimentação	Ventilador Ambiente	Ventilador dos Gases de Escape
	Inativo	OFF	OFF	OFF
Ignição	ON	ON	OFF	ON
Queima	OFF	ON	ON ou OFF	ON
Observação	OFF	OFF	OFF	OFF
Extinção	OFF	OFF	ON ou OFF	ON

Tabela 5.2 - Estado dos atuadores em função do estado do processo

Inicialmente, o sistema arranca no estado “Inativo”. Assim que é solicitado o início do processo, este passa automaticamente para o de “Ignição” onde os *pellets* são incinerados. Deste estado, quatro possíveis transições podem ocorrer: duas delas dão-se caso o utilizador ordene o término do processo de aquecimento, ou na situação em que a *NTC* seja desligada do controlador, obrigando o estado a transitar para “Extinção” (em todos os estados, existem estas possíveis transições) Especificamente nesta fase, caso o tempo decorrido desde o seu arranque ultrapasse os 15 minutos (representando falta de *pellets* na câmara de combustão se todos os atuadores estiveram a funcionar devidamente), a máquina de estado transita igualmente para a fase de “Extinção”. Por outro lado, se a temperatura dos gases de escape exceder os 60°C, então o controlador está preparado para transitar de estado para “Queima”. Importa referir que após a transição da fase de “Extinção” para a “Inativa”, por qualquer razão a temperatura no interior da câmara de combustão pode voltar a exceder os 50°C, devendo assegurar-se, nesse caso, que se transite de novo para a fase de “Extinção”. Esta transição é ainda assegurada no caso em que ocorra uma falha de energia (instantânea) e o processo de aquecimento esteja em andamento, já que a temperatura dos gases de escape se pode encontrar bastante elevada e interessa expulsar todo esse calor de dentro do aparelho. Quando tiver sido restabelecida a energia após uma falha mais prolongada, a máquina de estados encontrar-se-á na fase “Inativa”, sem efetuar qualquer ação.

De forma esquematizada, toda a informação sobre estas e outras transições possíveis entre estados está representada graficamente na Figura 5.2.

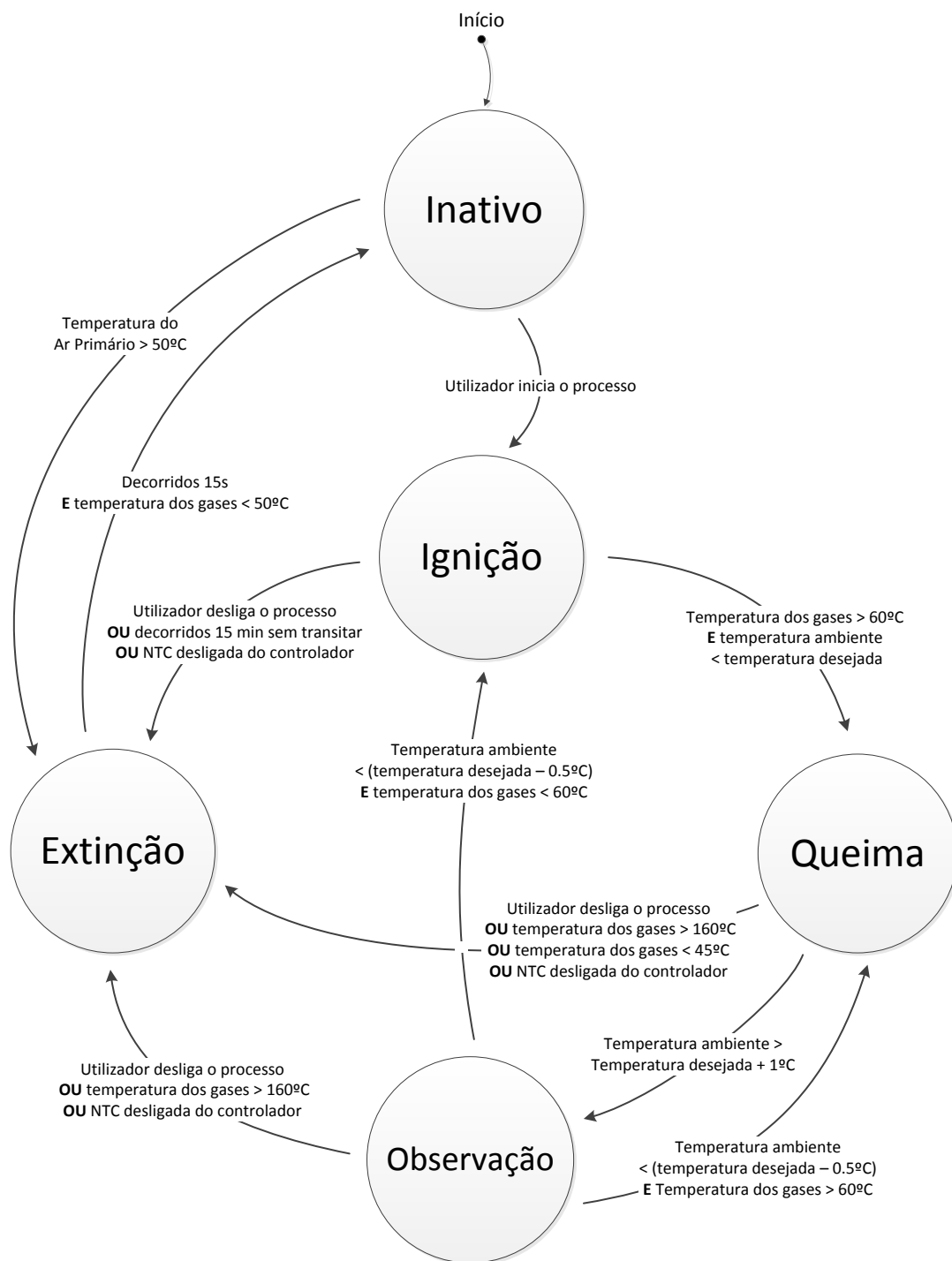


Figura 5.2 - Diagrama de transição de estados

5.1.2 - ARMAZENAMENTO E CONCEPTUALIZAÇÃO DOS PARÂMETROS DO SISTEMA

Os 56 *bytes* de RAM disponíveis na *RTC* foram utilizados para armazenar os parâmetros do sistema, que se encontram mapeados segundo a disposição apresentada na Tabela 8.1 (anexo).

Os parâmetros do sistema correspondem a variáveis com significado para o controlador e que servem de medida fixa de comparação para o bom funcionamento do programa que corre sobre ele.

Os parâmetros encontram-se divididos por categorias:

- Atuadores
- Temperaturas
- Crono termostato
- Cooperação
- Palavra-passe
- Calibração

Os parâmetros que dizem respeito aos atuadores estão diretamente associados ao processo de aquecimento, nomeadamente na fase de trabalho, em que cada um deles, exceto a resistência de ignição, possui regulação até cinco níveis. O primeiro nível corresponde ao grau de queima mais baixo, enquanto o quinto corresponde ao mais alto. Relativamente ao alimentador de *pellets*, cada nível corresponde a um certo intervalo de tempo em que este se encontra ligado, em percentagem, relativamente a um período de funcionamento, em segundos. Importa salientar que na fase de ignição este atuador não utiliza nenhum dos cinco níveis anteriores. Encontra-se inicialmente ligado durante um certo intervalo de tempo, passando posteriormente, a funcionar segundo um *duty-cycle* baixo enquanto a fase de trabalho não for atingida.

Nos ventiladores, cada nível corresponde a uma certa velocidade de rotação que é traduzida numericamente em tensão eficaz de 0 a 220 V.

No que às temperaturas diz respeito, destacam-se as do ar primário, como por exemplo, a máxima permitida, a de transição da fase de “Ignição” ou “Observação” para a fase de “Queima” e da fase de “Extinção” para a “Inativa”. Quanto ao ar secundário, ou ambiente, são estabelecidos limites mínimo e máximo de seleção da temperatura desejada e ainda um parâmetro para armazenamento da última escolha desta temperatura.

Para a funcionalidade do Crono termostato interessa guardar as horas e os minutos de início e fim do processo de aquecimento, assim como da temperatura de referência para cada programa, num total de três.

Ao nível da cooperação entre controladores, cada um deve possuir três parâmetros: o número identificador, que deve ser diferente de todos os outros, a temperatura a que entra em funcionamento e o seu nível de queima.

Para a Palavra-passe estão reservados 16 *bits*, ou seja, poderá corresponder a um valor numérico entre 0 e 65536.

Os parâmetros associados à calibração do sistema dividem-se em três: a temperatura inicial, final e tempo total, e traduzem algum grau de conhecimento do sistema em que a máquina está inserida.

Por último, importa frisar que o nível máximo de queima que é selecionado pelo utilizador deverá encontrar-se entre os cinco níveis descritos anteriormente. Por exemplo, no caso do valor selecionado ser 3, a máquina não vai poder funcionar num nível de queima de 4 ou 5.

À exceção dos parâmetros associados ao crono termostato, nível máximo de queima e temperatura desejada, que o utilizador normal pode gerir, todos os outros apenas podem ser acedidos e alterados pelo instalador/fabricante.

5.1.3 - CODIFICAÇÃO E DESCODIFICAÇÃO DA DATA E HORA ATUAL

Internamente, os registos internos da *RTC* estão codificados em *BCD*, tornando-se necessárias criar rotinas de codificação e decodificação.

A decodificação é necessária sempre que se pretenda ler a data e hora da *RTC*. Em *BCD*, os dígitos que constituem o número decimal são codificados pelo peso que representam. Uma vez que a gama máxima que se pode atingir no valor dos registos temporais desta *RTC* é de 0 a 99, referente aos dois dígitos menos significativos do ano (Tabela 5.3), uma possível conversão deste código para decimal será obter, em primeiro lugar, os dígitos mais e menos significativos (5.1 e 5.2) do número *BCD* e finalmente obter o número em decimal (5.3).

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	FUNCTION	RANGE
00h	CH	10 Seconds			Seconds				Seconds	00–59
01h	0	10 Minutes			Minutes				Minutes	00–59
02h	0	12	10 Hour	10 Hour	Hours				Hours	1–12 +AM/PM 00–23
		24	PM/ AM							
03h	0	0	0	0	0	DAY			Day	01–07
04h	0	0	10 Date		Date				Date	01–31
05h	0	0	0	10 Month	Month				Month	01–12
06h	10 Year				Year				Year	00–99
07h	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08h–3Fh									RAM 56 x 8	00h–FFh

Tabela 5.3 - Mapeamento de memória da RTC [48]

$$\text{Dígito dezenas} = 4 \text{ bits mais significativos (BCD)} \quad (5.1)$$

$$\text{Dígito unidades} = 4 \text{ bits menos significativos (BCD)} \quad (5.2)$$

$$\text{Número decimal} = \text{Dígito dezenas} \times 10 + \text{Dígito unidades} \quad (5.3)$$

A operação contrária, a de codificação para código *BCD*, utilizada na situação em que se pretende fazer a atualização da data e hora envolve igualmente dois passos.

O primeiro consiste em separar o número decimal nos dígitos das dezenas e das unidades, como na descodificação, ainda que não da mesma forma (5.4 e 5.5). O número *BCD* final é calculado em 5.6.

$$\text{Dígito dezenas} = \text{Número decimal} / 10 \quad (5.4)$$

$$\text{Dígito unidades} = \text{Número decimal} \% 10 \quad (5.5)$$

$$\text{Número decimal} = \text{Dígito dezenas} \times 16 + \text{Dígito unidades} \quad (5.6)$$

5.1.4 - DEVICE DRIVERS

Device drivers são rotinas específicas do *firmware* que interagem com componentes específicos do *hardware*. São responsáveis pela leitura ou escrita em registos, assim como pela resposta a interrupções, com a informação processada [58]. Dividiram-se os *device drivers* em dois tipos: para sensores e atuadores.

A leitura de um sensor deve ser individual e independente de qualquer outro. Conhecendo o *hardware* de acondicionamento de sinal e as suas especificações é possível realizar alguma manipulação matemática que traduza a grande física medida em outra que se pretende observar/medir.

Reproduzidos pelos sensores do sistema, os sinais analógicos são lidos através de três (num total de dezasseis) canais de 10 *bits* da ADC do PIC32. Cada canal pode ler até $2^{10} = 1024$ valores diferentes. Uma vez que a gama de variação da tensão lida situa-se entre 0 e 3.3V, isto significa que a resolução de leitura é $\frac{3.3}{1024} = 3.2\text{mV}$.

Devido ao erro associado ao processo de medida do sinal, a rotina de leitura de um certo canal da ADC, cujo esquema exemplificativo de funcionamento pode ser consultado na Figura 5.3, segue um determinado número de passos.

Em primeiro lugar são lidas dez amostras e salvaguardadas num *array* de dez elementos. Posteriormente é realizada uma pequena filtragem, em que são detetados os valores máximo e mínimo, que são eliminados. De seguida, é feita a média entre os restantes e convertida para tensão, em mV, segundo a expressão 5.7 ou, reduzindo o número de cálculos efetuados pelo microcontrolador, expressão 5.8.

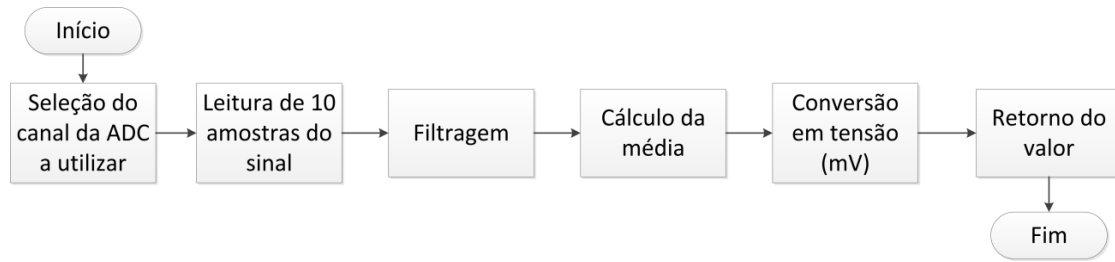


Figura 5.3 - Esquema da rotina de leitura da ADC

$$V_{mV} = VAL_{ADC} \cdot \frac{3.3 \cdot 1000}{1023} \quad (5.7)$$

$$V_{mV} = \frac{VAL_{ADC}}{0.31} \quad (5.8)$$

A leitura da temperatura ambiente é efetuada tendo em conta a diferença de potencial, que resulta do produto entre a resistência aos terminais da sonda *NTC* e da corrente de polarização definida, $107\mu A$ (Figura 4.11). Recorrendo à tabela existente em [50] que relaciona a sua resistência com a temperatura, traçou-se o Gráfico 5.1 com a evolução da temperatura em função da diferença de potencial. Utilizando uma regressão polinomial de ordem cinco, foi possível obter-se uma boa aproximação para esta relação (5.9). Previamente a este cálculo, a tensão lida em mV, deve ser convertida em V.

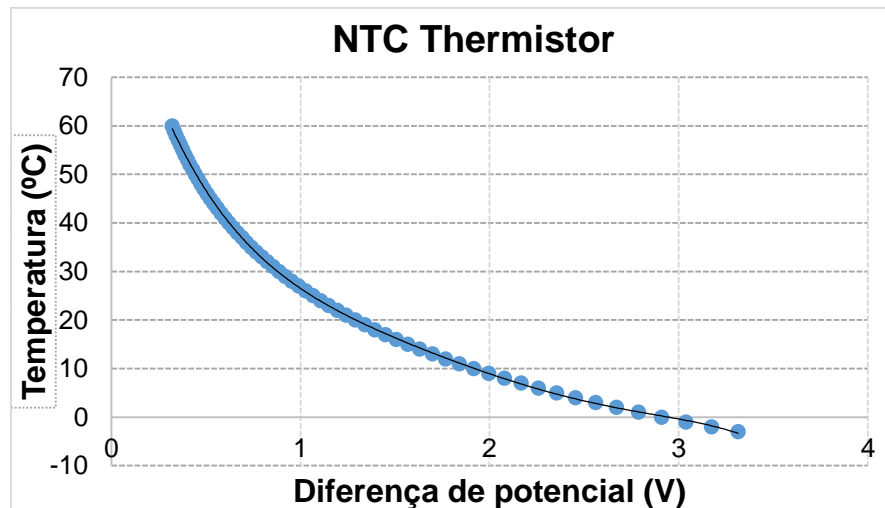


Gráfico 5.1 - Curva de Tensão-Temperatura da NTC e regressão polinomial de ordem 5

$$T_{Ambiente} = -1.416x^5 + 14.822x^4 - 60.777x^3 + 125.65x^2 - 147.53x + 95.762 \quad (5.9)$$

Segue-se o procedimento utilizado para a leitura da temperatura dos gases de escape por parte do termopar tipo K.

Depois de obtido o valor da tensão à saída do amplificador AD623N, é necessário dividi-lo pelo ganho antes de utilizar a expressão 5.10 [59], cujos coeficientes foram calculados tendo em conta a tabela presente em *NIST* [60].

$$T_{ref} = T_0 + \frac{(v - v_0) \left(p_1 + (v - v_0) \left(p_2 + (v - v_0) (p_3 + p_4(v - v_0)) \right) \right)}{1 + (v - v_0) \left(q_1 + (v - v_0) (q_2 + q_3(v - v_0)) \right)} \quad (5.10)$$

em que v corresponde à tensão aos terminais do sensor (em mV).

É importante frisar que o termopar não possui uma resposta linear em toda a gama de temperatura. Por essa razão, dividiu-se o cálculo da temperatura em dois troços para evitar descontinuidades e erros grosseiros [61]. Um primeiro dos -100°C aos 100°C e outro dos 100°C aos 400°C. Para cada um, é utilizado um conjunto de coeficientes associados a um polinómio diferente. No Gráfico 5.2 pode ser visualizada a evolução da temperatura calculada, em função da tensão aos terminais do sensor, utilizando os coeficientes da primeira gama para medir temperaturas até 400°C. Nota-se uma descontinuidade de cerca de 84°C por volta dos 5mV, além da temperatura final ficar apenas pelos 390°C. Utilizando os coeficientes associados aos dois troços (Gráfico 5.3), observam-se melhorias relativamente ao gráfico anterior. Comparando os dois métodos, o erro máximo associado à medição da temperatura utilizando apenas um troço foi de cerca de 45°C, enquanto utilizando dois troços de 0,02°C.

Por último, dado que a temperatura lida corresponde ao ponto em que o termopar se liga à placa, é necessário adicionar a temperatura da junção fria, mais concretamente, a temperatura da placa (5.11) – método de compensação por *software*.

$$T_{Gases\ Escape} = T_{ref} + T_{Placa} \quad (5.11)$$

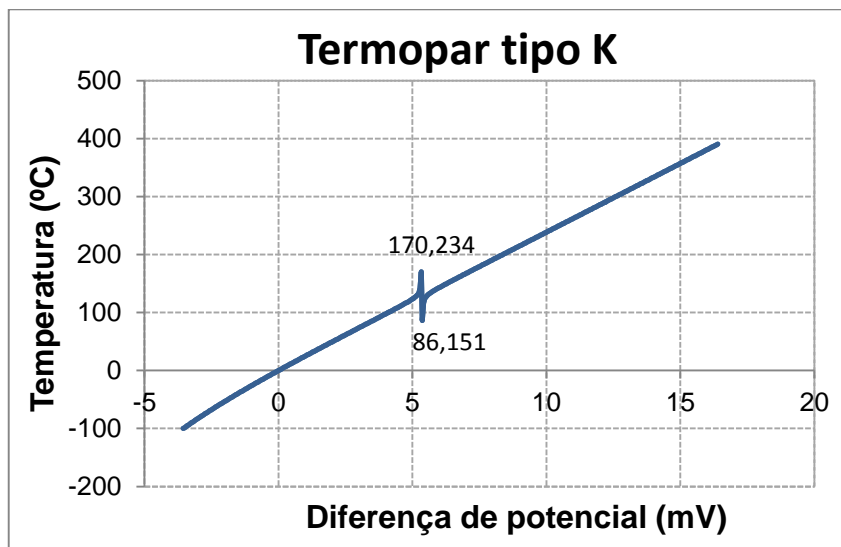


Gráfico 5.2 - Cálculo da temperatura utilizando apenas um troço

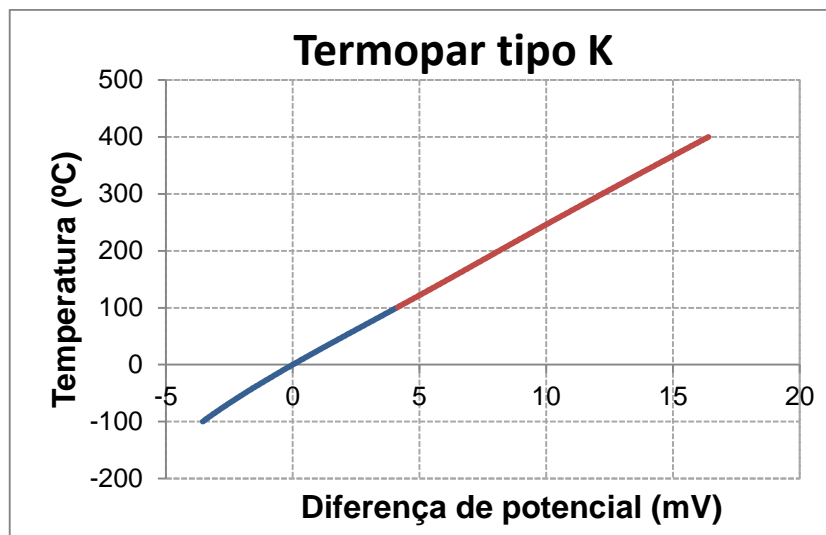


Gráfico 5.3 - Cálculo da temperatura utilizando dois troços

Tendo como principal objetivo realizar a compensação da junção fria do termopar, a leitura da temperatura da placa revela-se um importante aspeto a ter em conta. Para o efeito, utilizou-se o circuito integrado LM335 e o circuito de acondicionamento de sinal adequado (Figura 4.13). Uma vez que a gama de variação de temperatura considerada situa-se entre -20°C e 65°C e a respetiva gama de tensão à saída do amplificador é de 0.429V até aos 3.234V , é possível determinar-se uma relação linear (5.12) entre as duas grandezas através destes dois pontos. A reta obtida está representada no Gráfico 5.4.

$$T_{\text{Placa}} = 30.303 \cdot V_{\text{out}} - 33 \quad (5.12)$$

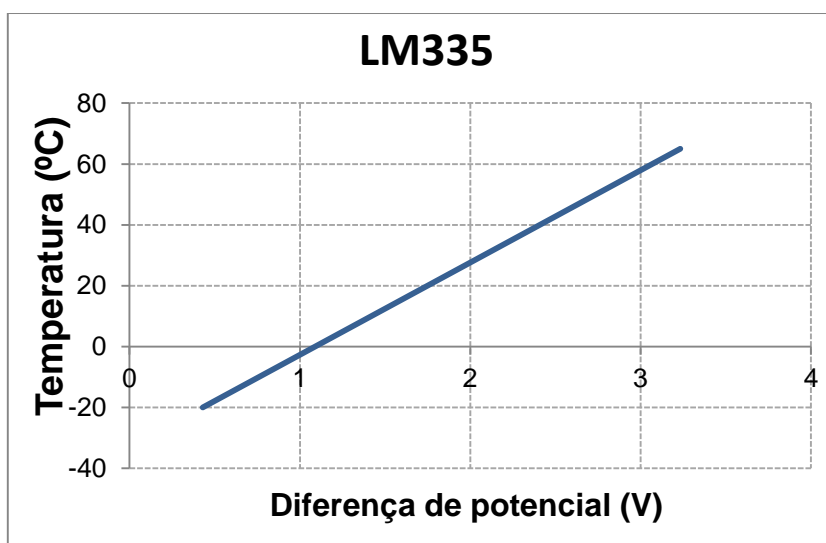


Gráfico 5.4 - Temperatura em função da tensão de saída do MCP6022

Os quatro atuadores elétricos descritos anteriormente possuem especificações próprias quanto ao seu modo de funcionamento, apesar da interface de potência ser igual para todos eles, já que trabalham utilizando a tensão da rede (220V).

A resistência de ignição é aquela cujo princípio de funcionamento é mais simples, tendo em conta a especificação da máquina de estados do processo de aquecimento. Esta funciona apenas com controlo ON/OFF, ou seja, aos seus terminais está aplicado 0V ou 220V em qualquer instante. Assim, a saída digital do microcontrolador, destinada à atuação deste dispositivo, deve ser colocada no valor lógico “0” quando se pretende desligá-la, ou “1” na operação contrária (Figura 4.18).

O motor de alimentação possui um princípio de funcionamento muito semelhante. Contudo, conforme anteriormente dito, o controlo especificado para este motor precisa de ter em conta o período de funcionamento e a percentagem desse período em que este atuador se deve encontrar ligado, não existindo controlo de velocidade de rotação. Sempre que é dada ordem para se ligar, ativando a respetiva saída digital do microcontrolador, a contabilização do tempo é reiniciada. Esta contagem é efetuada por um *Timer* que dispara uma interrupção a cada 125 milissegundos. Decorridos oito disparos seguidos, significa que decorreu um segundo e a variável que mede o tempo em que este atuador se encontra ligado é incrementada.

Os ventiladores ambiente e de escape são controladas de forma idêntica uma à outra, mas diferente dos dois anteriores. Para controlo da velocidade de rotação de ambas foi utilizada a técnica de regulação do ângulo de disparo, o que implica uma sincronização com a tensão da rede elétrica. Ou seja, é necessário o controlador conhecer os instantes de passagem por zero, para determinar a partir daí os instantes de disparo. Essa sincronização é estabelecida pelo circuito da Figura 4.16 que utiliza o pino RD11 de entrada do microcontrolador. Sempre que neste pino ocorrer uma transição ascendente do nível lógico é disparada uma rotina de interrupção externa. A partir desse momento, é ativado e reiniciado um *Timer* por cada ventilador, caso estejam ligadas, que conta o tempo relativamente ao instante de passagem por zero da rede (Figura 5.4).

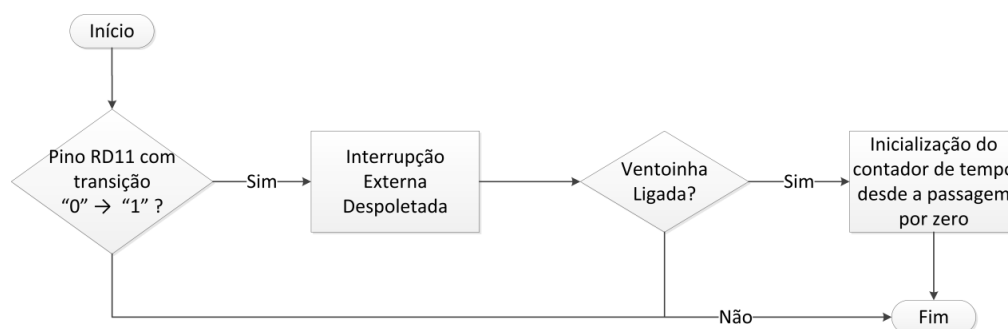


Figura 5.4 - Fluxograma de sincronização com a tensão da rede elétrica

A frequência da onda sinusoidal da rede elétrica é de 50 Hz, o que significa que de 10 em 10 milissegundos esta atinge o valor zero. Por esta razão, a frequência do *Timer* deve ser de 100 Hz para impedir que a contagem do tempo não ultrapassasse mais do que uma arcada da onda.

Recorrendo à Tabela 5.4, podem observar-se os valores de instantes de disparo associados à velocidade máxima e mínima de ambas os ventiladores. O facto da velocidade máxima dos ventiladores só ser conseguida alguns milissegundos após a passagem por zero deve-se ao *TRIAC* ser, de facto, controlado por corrente, havendo um desfasamento entre esta e a tensão, devido ao cariz de carga indutiva.

Ventilador	Instante de disparo mínimo	Instante de disparo máximo
Ambiente	4.1s	6.1s
Gases de Escape	3.7s	6.5s

Tabela 5.4 - Instantes de disparo relativos aos extremos de velocidade dos ventiladores

Sendo utilizados valores de tensão para caraterizar cada um dos cinco níveis, estabeleceu-se que o valor mínimo de instante de disparo corresponde a 220V e o valor máximo a 0V, o gráfico obtido para cada um dos ventiladores (Gráfico 5.5 e Gráfico 5.6) representa a regressão linear realizada sobre estes dois pontos e cujas respetivas equações se encontram em 5.13 e 5.14.

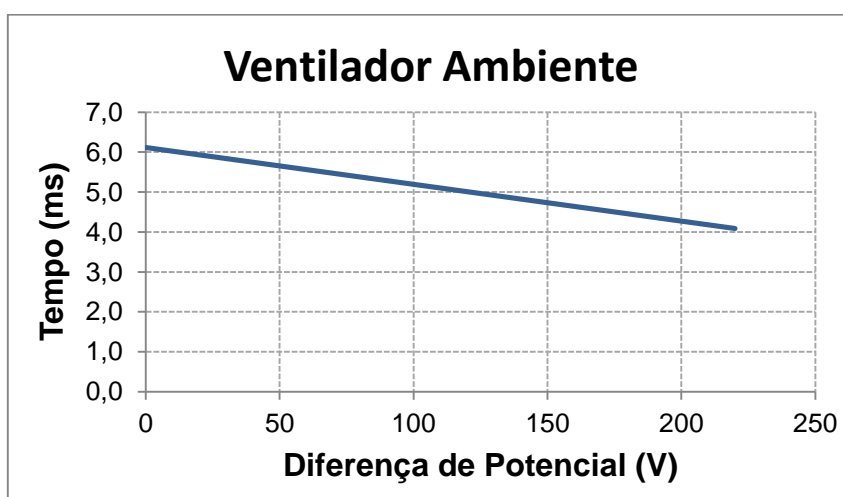


Gráfico 5.5 - Relação entre o Instante de disparo e a tensão aplicada ao VA

$$t = -0.0127v + 6,5139 \quad (5.13)$$

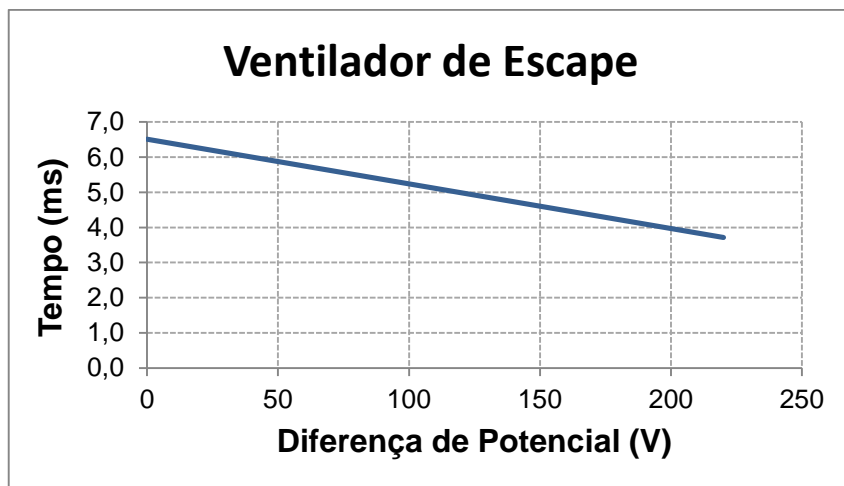


Gráfico 5.6 - Relação entre o Instante de disparo e a tensão aplicada ao VE

$$t = -0.0092v + 6,1138 \quad (5.14)$$

Sempre que o respetivo *timer* atingir o valor de instante de disparo, pára o seu próprio funcionamento e cria o disparo do *TRIAC*, ativando a respetiva saída do microcontrolador. O disparo é concluído, ou seja, a saída é colocada no valor lógico “0” no programa principal.

5.1.5 - FREQUÊNCIA DE AMOSTRAGEM DAS TEMPERATURAS

A taxa da aquisição de amostras para a leitura dos três tipos de temperaturas consideradas é definida através de um único *timer* que gera uma RSI a cada 125 milissegundos. Dentro da rotina existe uma variável contadora para cada uma das temperaturas.

Devido ao facto da temperatura ambiente ter uma evolução relativamente lenta ao longo do tempo, cada amostra recebida (de 125 em 125 milissegundos) não é imediatamente utilizada para atualizar a variável que armazena a temperatura ambiente. Assim sendo, a cada 30 segundos é calculada a média sobre as 30 amostras adquiridas durante esse intervalo de tempo e só aí é que o novo valor é amostrado.

Relativamente à temperatura dos gases de escape, a sua resposta é mais rápida que a anterior, notando-se essa diferença principalmente quando a fase do processo de aquecimento é a de “Ignição” e porque o volume de ar é inferior (câmara de combustão). Por esta razão, optou-se por utilizar uma frequência de amostragem de uma vez em cada segundo (1 Hz), sendo calculada a média sobre as 8 amostras adquiridas durante esse intervalo de tempo. Devido ao facto desta necessitar de

compensação da junção fria, a frequência de amostragem da temperatura da placa deve estar sincronizada com a anterior. Esta informação foi recolhida pela Tabela 5.5.

Temperatura	Frequência de amostragem
Ambiente	$\frac{1}{30}$ Hz
Gases de Escape	1 Hz
Placa	1 Hz

Tabela 5.5 - Frequências de Amostragem de cada tipo de temperatura

5.1.6 - CRIAÇÃO DE PRIMITIVAS GRÁFICAS E “WIDGETS”

No que se refere aos elementos gráficos possíveis de ser reproduzidos no ecrã do *LCD*, consideram-se dois tipos: os mais simples, as primitivas gráficas e os mais complexos que podem ser criados com combinações dos primeiros, os “*widgets*”.

Entenda-se como primitiva gráfica qualquer elemento possível de ser suportado pela lista de displays do *LCD* sem o auxílio de outro, como por exemplo: pontos, linhas, retângulos, ou mapa de *bits*. A partir destes podem criar-se caracteres individuais, texto, números, botões, relógios analógicos e digitais, velocímetros, barras de progresso, barras deslizantes para controlo de volume, barras de seleção, gradientes de cor, etc.

Por exemplo, para se criar um botão basta invocar o comando *CMD_BUTTON* e especificar os seus parâmetros de entrada: coordenadas horizontal e vertical do canto superior esquerdo (“*x*” e “*y*”) do botão, em pixéis, o tamanho dos caracteres de texto a mostrar (“*font*”), opções de alinhamento do texto, com ou sem 3D, entre outros (“*options*”) e, por último, o texto propriamente dito a introduzir na área delimitada pelo botão (“*s*”). Este e outros “*widgets*” podem ser acedidos através da biblioteca de suporte de elementos gráficos compatível com o FT800.

5.1.7 - CRIAÇÃO DE UM ECRÃ

Um ecrã representa o resultado da conjugação de um ou vários objetos gráficos visíveis. Para se criar um ecrã é necessário respeitar uma sequência lógica de comandos a enviar ao motor gráfico:

- 1 - Início um novo ecrã (`cmd(CMD_DLSTART)`)
- 2 - Definição da cor de fundo (`cmd(CLEAR_COLOR_RGB(R, G, B))`)
- 3 – Limpeza de ecrã (`cmd(CLEAR(1, 1, 1))`)
- 4 - Construção do(s) objeto(s) gráfico(s)

5 - Sinalização do fim da criação do ecrã (`cmd(DISPLAY())`)

6 - Ativação do novo ecrã na área de display (`cmd(CMD_SWAP)`)

No caso de um destes passos ser ignorado, excetuando o quarto, o ecrã não se torna visível.

5.1.8 - TOUCH

Através da funcionalidade de *Touch* no *LCD*, torna-se mais fácil a interação com o utilizador através de um simples toque na área correspondente ao objeto gráfico (botão, retângulo, círculo, etc.) criado com essa funcionalidade.

Para criar um objeto gráfico que reproduza um sinal sempre que tocado utiliza-se um comando já pré-programado na memória de armazenamento de comandos gráficos do VM800C: *CMD_TRACK*. Sequencialmente são descritos abaixo os passos necessários para acrescentar esta funcionalidade a um certo objeto gráfico:

- 1 – Definição de um número identificador para o elemento ao qual se pretende introduzir a funcionalidade de toque - *TAG*.
- 2 - Criação do elemento gráfico e escolha o tipo de monitorização de toque a introduzir. Existem dois: linear ou rotativo
- 3 – Utilização do comando *CMD_TRACK* para estabelecer a funcionalidade de toque ao elemento criado.
 - Monitorização de toque Linear – é necessário especificar as coordenadas horizontais (“x”) e verticais (“y”), em pixéis, do canto superior esquerda do elemento, o seu comprimento (“w”) e altura (“h”) e por fim introduzir a *TAG* anteriormente definida.
 - Monitorização de toque Rotativo – é necessário especificar as coordenadas horizontais (“x”) e verticais (“y”), em pixéis, do centro do elemento, o valor “1” nos campos “w” e “h” e por fim introduzir a *TAG* anteriormente definida.

A partir deste momento o elemento gráfico monitorizado responde ao toque com o valor da *TAG* atribuída. Este valor pode ser lido nos 8 *bits* menos significativos do registo *REG_TOUCH_TAG* da categoria de registos do mecanismo gráfico do FT800.

5.1.9 - GESTÃO DE ECRÃS

A forma como os ecrãs interagem uns com os outros e se sincronizam com todos os processos que se encontram a executar em simultâneo será descrita de seguida.

No total foram elaborados vinte e dois ecrãs que transitam entre si por meio de botões, retângulos, imagens ou outros elementos gráficos.

A unidade central de processamento necessita de saber, em primeiro lugar, o ecrã ativo a cada instante e, posteriormente, em qual dos elementos gráficos o utilizador toca para a transição se dar de forma correta. Em certas situações pode ser ainda necessário conhecer o ecrã que foi ativado imediatamente antes do atual. Na Figura 5.5 encontra-se graficamente toda a gestão de ecrãs e respetivas transições.

O primeiro ecrã (Figura 8.4) é lançado automaticamente assim que o controlador é inicializado. Todos os outros vão sendo colocados ativamente no *LCD* através de determinadas combinações de toques nos elementos gráficos certos.

Para a gestão de todos os ecrãs criou-se uma rotina que lê continuamente o valor da *TAG*, cujo objeto gráfico esteja a ser tocado pelo utilizador. Sempre que um novo ecrã seja mostrado no *LCD*, por ter sido despoletada essa ação, através de um toque num botão, por exemplo, o controlador terá conhecimento desse acontecimento e guardará essa informação. O valor da *TAG* poderá ir de 0 a 255, ou seja, à partida seria possível ter-se 256 objetos gráficos diferentes com funcionalidade de toque no mesmo ecrã. Contudo, há que considerar o caso em que nenhum elemento esteja a ser tocado. Para essa situação foram definidos os valores da *TAG*, por predefinição, de 0 ou 255. Nenhuma ação é despoletada nesses casos e, portanto, são possíveis de monitorizar 254 objetos gráficos num mesmo ecrã.

A título de exemplo podem-se considerar os três primeiros ecrãs, correspondentes às Figuras 8.4, 8.5 e 8.6. Facilmente se percebe que o ecrã de “Processo ativo” resulta do toque no botão “*START*” do ecrã de “Processo inativo” e vice-versa caso o botão “*STOP*” seja tocado no ecrã de “Processo ativo”. De ambos os ecrãs se podem seguir todos os outros, como por exemplo de amostragem de temperaturas, alteração de parâmetros do sistema, etc. O terceiro ecrã aqui considerado pode ser acedido a partir de qualquer um dos dois primeiros através do toque na imagem com o símbolo de uma chama. Nele, o valor do nível máximo de queima do processo pode ser ajustado desde 1 até 5 através dos botões “+” e “-”, sendo que a cada nova alteração é invocado o mesmo ecrã com o resultado dessa modificação e o novo valor é guardado na memória do controlador. Para que se consiga voltar a um dos ecrãs iniciais, basta apenas que o botão “*Back*” seja premido.

Existem também alguns casos em que a transição de um ecrã para outro se dá sem o auxílio de qualquer toque. Um exemplo disso pode ser observado quando o processo de aquecimento se encontra na parte terminal da fase de “Extinção” (após um “*STOP*” ou excesso de temperatura dos gases, por exemplo). Assim que a temperatura de transição para a fase “Inativa” for atingida, o processo é desligado e o *LCD*

acompanha essa mudança colocando imediatamente ativo o ecrã de processo inativo (Figura 8.4).

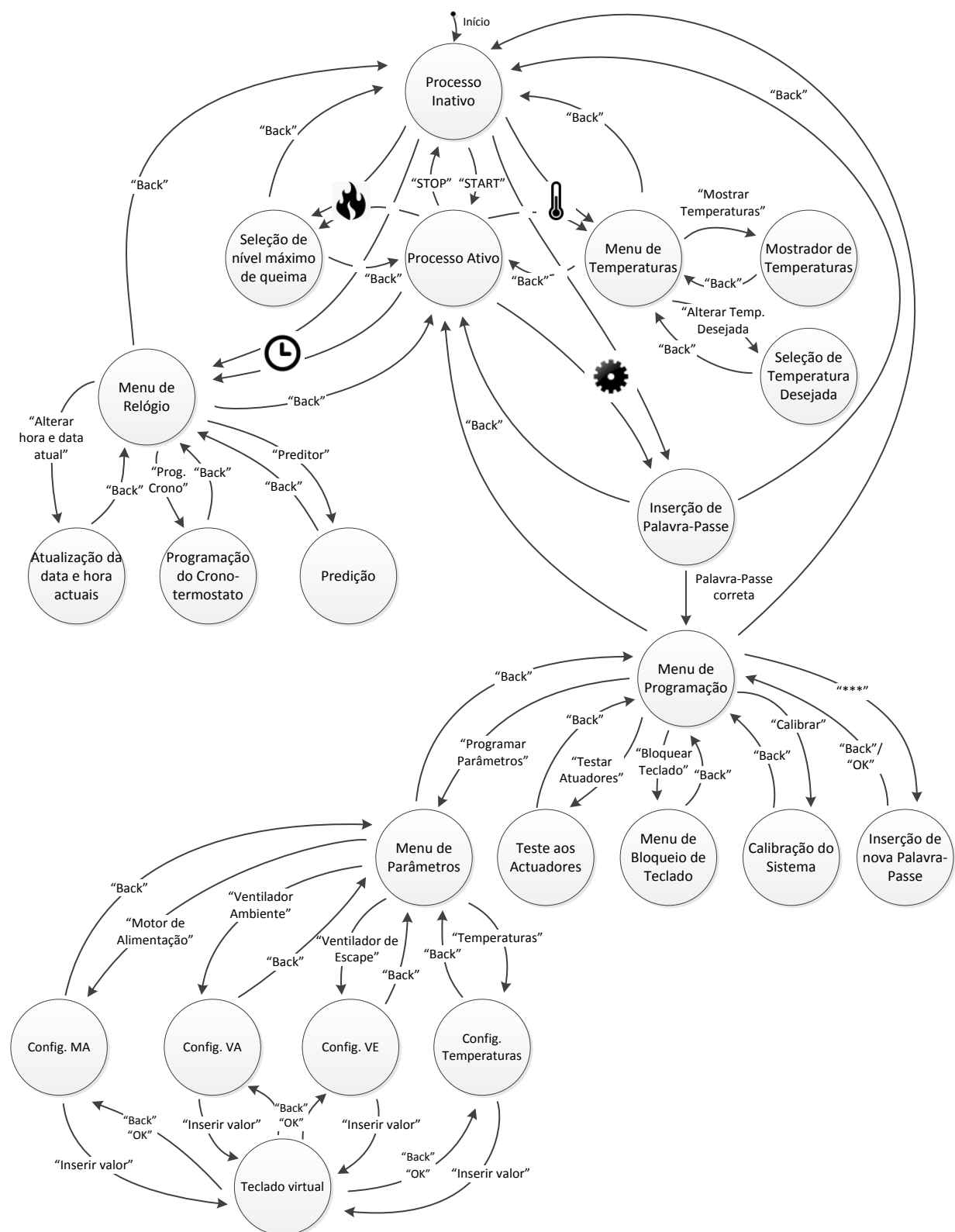


Figura 5.5 - Diagrama de transição entre ecrãs

5.1.10 - PROGRAMAÇÃO DA INTERFACE COM O LCD

Com base em critérios de simplicidade de análise, compreensão e legibilidade do código, a estrutura de programação utilizada assenta em três camadas lógicas principais (Figura 5.6):

- Camada 1 (Comunicação *SPI*)
- Camada 2 (Gestão gráfica e de áudio)
- Camada 3 (Ecrãs ou aplicações)

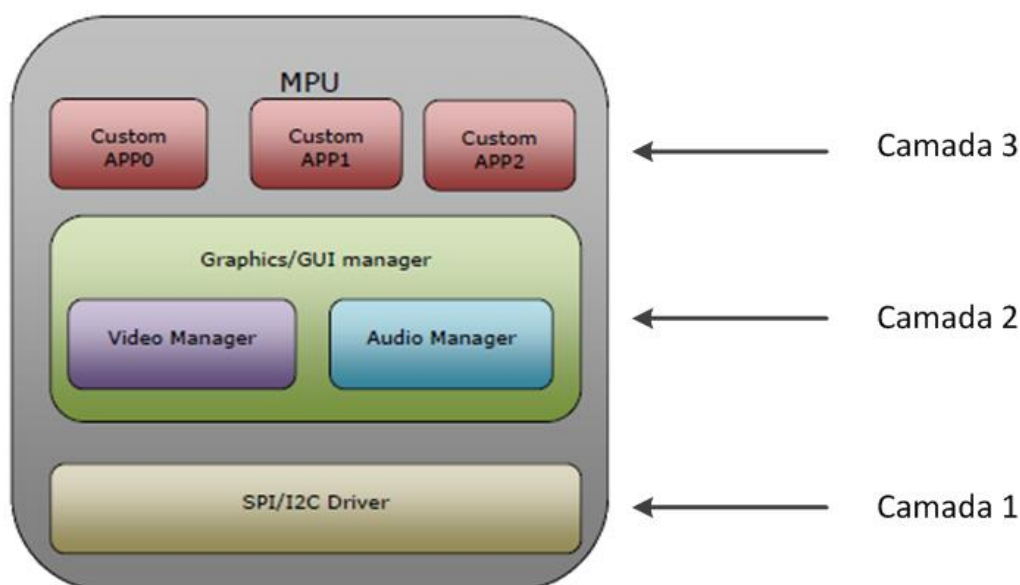


Figura 5.6 - Camadas lógicas para interface com o FT800

A primeira camada situa-se entre os mecanismos de gestão gráfica e o *hardware* propriamente dito. Preferiu-se usar *SPI*, cuja descrição do protocolo se encontra no anexo 8.3, em detrimento de *I2C* pela maior simplicidade de implementação. Nesta camada estão criadas todas as rotinas de acesso ao barramento: leitura e escrita de palavras com diferentes comprimentos (8, 16 e 32 *bits*). Na escrita, entenda-se ainda um comando como um conjunto de palavras de comprimento não necessariamente igual e que concretizem um determinado elemento ou funcionalidade gráfica. Na camada de gestão gráfica e de áudio estão presentes todas as rotinas responsáveis pela transferência adequada de dados para o *buffer* de comandos gráficos (Figura 5.7). Este *buffer* com a capacidade para 4K *bytes* de armazenamento está contido na memória do FT800 e funciona segundo um mecanismo em anel, reencaminhando os dados para o motor gráfico. O endereço inicial onde está contido é identificado pela macro *RAM_CMD*, cujo endereço corresponde a 10800h. Quando vazio, os registos *REG_CMD_WRITE* e *REG_CMD_REG* são iguais, ou seja, apontam para o mesmo endereço de memória. À medida que comandos/palavras são nele depositados, o

valor de *REG_CMD_WRITE* vai sendo incrementado. Quando o motor gráfico detetar uma determinada diferença entre ambos os endereços, associados ao objeto gráfico (botão, por exemplo), processa os dados e ambos os ponteiros voltam a apontar para o mesmo local. Sempre que forem depositados comandos no *buffer*, deve ser verificado previamente o espaço livre disponível para que caso não haja, a operação não ser permitida. Nesta camada já é possível construírem-se objetos gráficos mas ainda não poderão ser visualizados no ecrã.

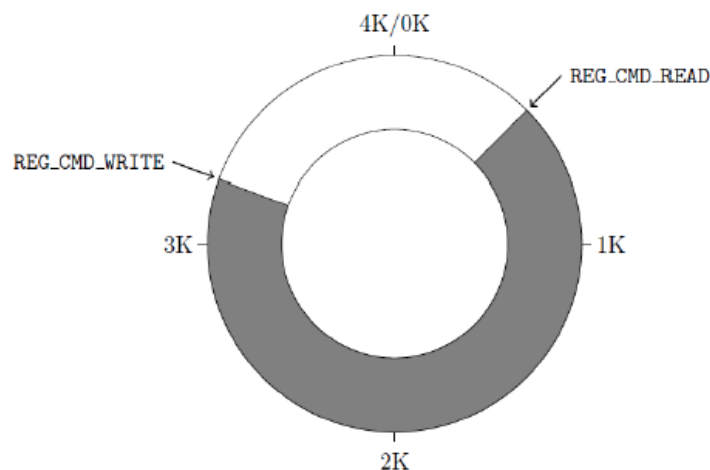


Figura 5.7 - *Buffer* de comandos gráficos

Na terceira camada, a de mais alto nível, o programador constrói o ecrã gráfico preocupando-se apenas com os elementos (criados na camada anterior) que nele pretende adicionar (botões, palavras, imagens, etc) e/ou fazer a gestão de vários ecrãs (aplicação).

5.1.11 - PROGRAMAÇÃO DO CRONO TERMOSTATO

Foram criados três programas horários diferentes que podem ser livremente manipulados pelo utilizador. Contudo, existem algumas considerações a ter em conta, como por exemplo o fato do controlador dever impedir que a hora de início do processo seja posterior à de fim, notificando e obrigando o utilizador a alterar uma delas. No caso em que um programa esteja agendado e tenha sido solicitada a sua alteração, para outro, o utilizador deve ser confrontado com uma mensagem de confirmação desse desejo. Se, por algum motivo, o processo de aquecimento estiver em andamento no instante em que o programa agendado for iniciado, esta solicitação fica sem efeito. O ecrã criado para programação do crono termostato no *LCD* (Figura 8.12) possui:

- Dois botões, que correspondem a setas, para navegação entre os três programas;
- Quatro botões para seleção da hora e do minuto de início e fim do processo de aquecimento;
- Dois botões para seleção da temperatura ambiente desejada.

5.1.12 - CALIBRAÇÃO

Para se poder usufruir da funcionalidade de predição, é necessário previamente obter-se algum conhecimento do sistema em que a máquina se encontra instalada. Esta ação de identificação de sistema foi denominada de calibração.

Para ser dado o seu início, o programador define um *setpoint* de temperatura, que deve ser atingido no processo de aquecimento e clica no botão “OK” do respetivo ecrã (Figura 8.24). Quando a temperatura atingir o *setpoint*, é contabilizado o intervalo de tempo a partir do qual a temperatura ambiente excedeu meio grau relativamente à temperatura inicial até esse momento. Ao mesmo tempo é armazenado esse valor na memória do controlador e mostrado no ecrã, assim como as temperaturas inicial e final.

5.1.13 - PREDIÇÃO

Conhecidas as temperaturas ambiente inicial e final, assim como o tempo útil decorrido da calibração, é possível calcular o tempo necessário ligar previamente o controlador antes da hora programada.

Dada a evolução da temperatura do compartimento em função do fluxo de ar quente enviado pelo recuperador de calor, o sistema pode ser aproximado por um modelo de primeira ordem, com tempo morto, como representado na Figura 5.8, em termos ideais. O modelo é simplista por várias condicionantes como a ausência de sensores de temperatura no exterior do compartimento, as variações do sistema provocadas pela abertura e fecho de portas e janelas e a presença ou ausência de pessoas no seu interior. A utilização de um modelo mais complexo não traria grandes vantagens nesta fase de desenvolvimento, já que a precisão desta funcionalidade não é um fator decisivo e seria necessário um gasto de tempo elevado na sua conceção.

Sabendo a temperatura inicial e final e ainda o tempo decorrido da calibração, é possível encontrar-se uma aproximação linear, que relacione a diferença entre a temperatura desejada e a temperatura ambiente com o tempo necessário ligar antes da hora programada (5.15).

A unidade de tempo aqui considerada é o minuto.

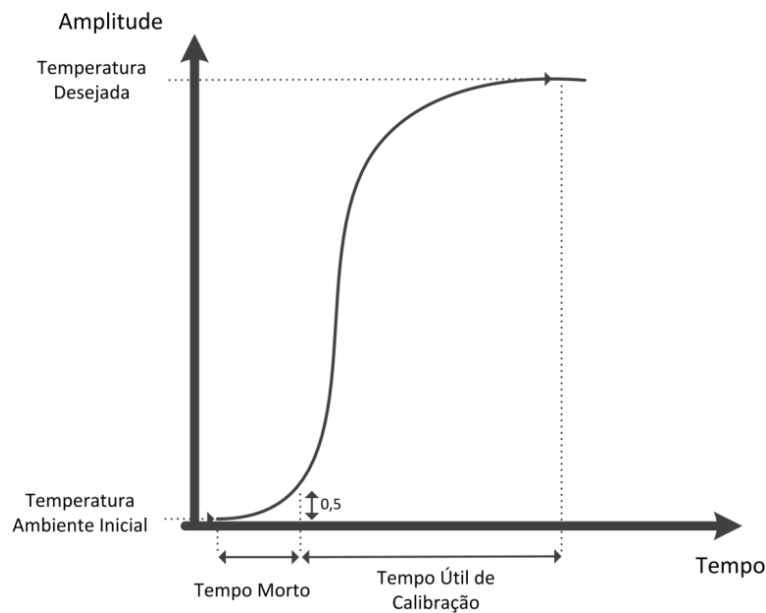


Figura 5.8 - Evolução ideal da temperatura ambiente durante a calibração

$$Tempo\ prévio = \frac{TUC}{T_{FinalC} - T_{InicialC} - 0,5} \cdot (T_{Desejada} - T_{Amb}) + Tempo\ morto \quad (5.15)$$

Depois de calculado o tempo necessário ligar o processo de aquecimento antes da hora programada, este deve ser-lhe descontado.

Uma vez que a hora programada vem expressa em horas e minutos, deve ser realizada uma pré-conversão (5.16 e 5.17) do tempo calculado em 5.15, visto este se encontrar somente em minutos. Posteriormente deve ser calculada a hora (hh) e o minuto (mm) de início do processo de aquecimento. No caso da hora calculada em (5.16) ser inferior à programada, a hora a ligar é obtida pela diferença entre ambas (5.18), enquanto na situação contrária deve ser descontada essa diferença, em módulo, relativamente às 24 horas (5.20). No caso dos minutos calculados em 5.17 serem inferiores ao programado, o minuto a ligar é calculado igualmente pela sua diferença (5.19), enquanto se forem superiores deve ser descontada a diferença, em módulo, sobre 60 minutos (5.21) e ainda decrementada a hora de uma unidade.

$$Horas\ a\ descontar\ (hh) = \frac{Tempo\ calculado}{60} \text{ (parte inteira)} \quad (5.16)$$

$$Minutos\ a\ descontar\ (mm) = Tempo\ calculado \% Horas\ a\ descontar \quad (5.17)$$

$$Hora\ a\ ligar = Hora\ programada - Horas\ a\ descontar \quad (5.18)$$

$$Minuto\ a\ ligar = Minuto\ programado - Minutos\ a\ descontar \quad (5.19)$$

$$Hora\ a\ ligar = 24 - |Hora\ programada - Horas\ a\ descontar| \quad (5.20)$$

$$Minuto\ a\ ligar = 60 - |Minuto\ programado - Minutos\ a\ descontar| \quad (5.21)$$

O ecrã criado para o preditor no *LCD* (Figura 8.13) possui:

- Quatro botões para seleção da hora e minuto a que o utilizador pretende ter a casa aquecida;
- Dois botões para seleção da temperatura ambiente desejada.

5.1.14 - COMUNICAÇÃO GATEWAY-CONTROLADORES

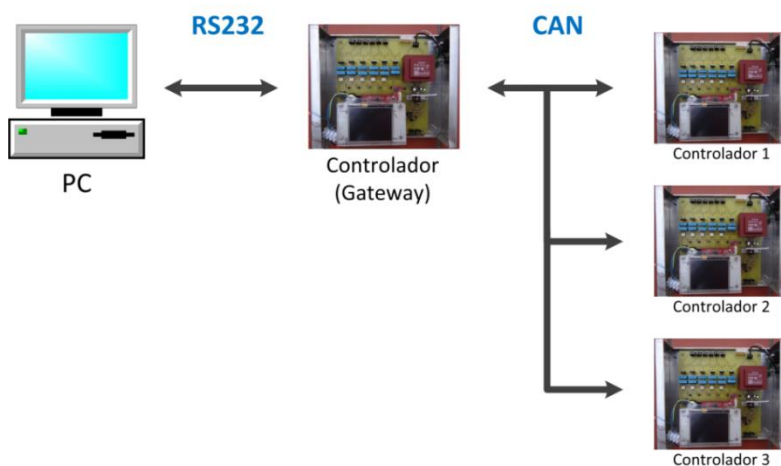


Figura 5.9 - Esquema de ligação *PC-gateway-controladores*

Como referido anteriormente, a funcionalidade de cooperação de controladores requer que várias caldeiras desempenhem ações de forma autónoma para um fim comum – o aquecimento de águas sanitárias. O protocolo de comunicação utilizado (*CAN 2.0B*), tem como base a utilização de mensagens e encontra-se descrito no anexo 8.3.

Cada um dos controladores envia uma mensagem de 2 em 2 segundos com o valor dos seus três parâmetros (identificador, temperatura e nível de queima). No formato da mensagem estão reservados 8 *bytes* destinados para o campo de dados, dos quais apenas os três primeiros se encontram preenchidos (Figura 5.10). Importa ainda frisar que o número identificador do destinatário da mensagem, que neste caso será o *gateway*, deve obrigatoriamente ser enviado na mesma trama, para poder ser efetivamente recebida apenas por ele. Definiu-se que os valores 0 ou 255 apenas podem ser associados a ele, sendo os restantes válidos para os controladores.

ID	Temperatura	Nível de Queima					
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7

Figura 5.10 – Campo de dados da trama enviada por cada controlador

Para receção da mensagem, o *gateway* reserva três *arrays* de dez elementos, para armazenar os parâmetros de cada controlador. Sempre que receber uma mensagem,

verifica se o identificador é válido (0 ou 255) e se corresponde a um novo identificador, ou seja, que não exista no *array* respetivo. Nesse caso, insere-o na lista e ordena-a, colocando os restantes dois parâmetros no mesmo lugar nos respetivos *arrays*. Finalmente, é reinicializado o contador respeitante ao ID recebido, seja ele novo ou não. Este contador individual para cada um dos controladores presentes no barramento caso ultrapasse 4 segundos significa que ocorreu um *time-out* e automaticamente é apagado o registo do controlador em questão.

A transmissão de uma mensagem por parte do *gateway* acontece sempre que seja solicitada a alteração de um ou mais parâmetros de um determinado controlador. O campo de dados da mensagem enviada deve possuir os novos parâmetros do controlador solicitado, assim como o identificador do *gateway* (Figura 5.11). De salientar que o identificador antigo do controlador deve ser sempre salvaguardado, já que este não teria forma de reconhecer a mensagem. O controlador ao receber a mensagem e reconhecer o remetente, altera internamente o valor dos parâmetros que não correspondam com os seus atuais. A partir desse momento, estará pronto para enviar os seus novos parâmetros ao *gateway*.



Figura 5.11 – Campo de dados da mensagem enviada pelo *gateway*

5.1.15 - ESTRUTURA DA FUNÇÃO MAIN

O programa principal que corre sobre o microcontrolador é denominado de função *MAIN*, onde são declaradas e inicializadas variáveis, configurados registos, processados dados, etc. Por ordem cronológica são especificadas as operações incluídas no programa:

- 1 - Declaração e inicialização de variáveis;
- 2 - Configuração e inicialização dos pinos de entrada e saída;
- 3 - Desativação de todas as interrupções;
- 4 - Configuração da rotina de interrupção externa;
- 5 - Configuração dos *Timers*;
- 6 - Configuração dos módulos de comunicação;
- 7 - Configuração dos registos do *LCD*;
- 8 - Leitura dos parâmetros do sistema;
- 9 - Cálculo das temperaturas iniciais;
- 10 - Cálculo da hora atual;

- 11 - Lançamento do ecrã inicial;
- 12 - Ativação das interrupções;
- 13 - Ciclo infinito de operação.

Dentro do último ponto é onde se desenrola toda a atividade de processamento que dá “vida à máquina”. De forma esquemática, o fluxograma do ciclo infinito de operação encontra-se representado na Figura 5.12.

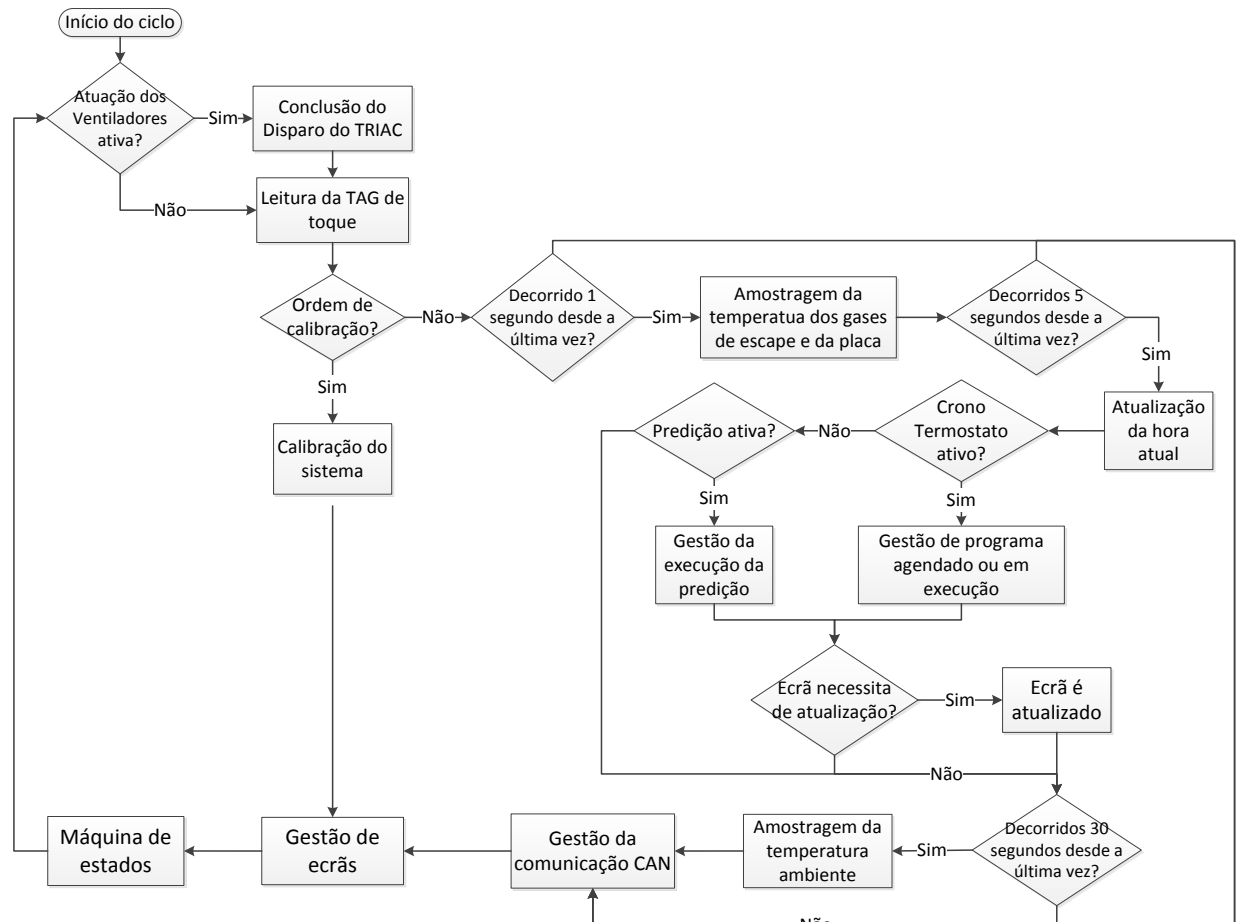


Figura 5.12 - Fluxograma do ciclo infinito da função *MAIN*

5.2 SOFTWARE PARA PC

Utilizando a ferramenta *NetBeans* IDE, foi criada uma aplicação para *PC*, em *JAVA*, com o intuito de ser utilizada pela *ECOAMA*, em função da máquina a que se destina, para instalação, configuração de parâmetros e depuração de avarias. Nesta dissertação, a aplicação desenvolvida destina-se apenas para um recuperador de calor, tendo em conta que é o único que pode ser testado nas instalações da empresa. Foi escolhida esta linguagem de programação por ser universal e portátil (pode correr

em qualquer máquina, desde computadores, telemóveis, *tablets*, etc, a partir de uma máquina virtual, *Java Virtual Machine*).

A aplicação permite ações de teste aos atuadores, configuração de parâmetros do sistema, programação do crono termostato, monitorização do processo de aquecimento, assim como do estado do barramento *CAN* em tempo real. Com o objetivo de facilitar a instalação e manutenção de um recuperador de calor no interior de uma casa por parte do fabricante, procurou-se que esta fosse o mais intuitiva possível e de depuração rápida de possíveis problemas com o *hardware*.

Para se utilizar a aplicação desenvolvida, o computador deve possuir uma porta série RS232 ou USB, desde que se utilize um conversor adequado, e esta encontrar-se ligada ao controlador.

5.2.1 - COMUNICAÇÃO COM A PORTA SÉRIE

A porta série foi escolhida como elo de ligação entre o computador e o controlador. Sendo a interface de comunicação bem definida (RS232) existem parâmetros de configuração e/ou de trama que ambos devem respeitar. São eles, principalmente, a *baud rate*, ou seja, a taxa de transmissão de símbolos por segundo, o número de *bits* de dados, de *stop bits* e o tipo de paridade, podendo ser utilizados outros.

Para o microcontrolador poder aceder à porta série utilizou-se o módulo 2 da *UART* disponibilizado pelo PIC32. De entre várias configurações do módulo destacam-se os quatro principais referidos acima:

- *Baud rate*: 115200 *bits/s*
- *Data bits*: 8
- *Stop bits*: 1
- Paridade: Sem paridade ('N')

Relativamente à leitura de informação da porta série, essa tarefa fica ao encargo da rotina de interrupção (Figura 5.13) que é chamada sempre que um novo carácter lhe chega. Depois de inicializado, o controlador não deve reconhecer nenhum outro comando que não seja um dos utilizados para a conexão com a aplicação. Quando esta estiver estabelecida, a rotina já o poderá fazer. Sempre que um comando tiver sido reconhecido, é executada a resposta à solicitação e o *array* utilizado para armazenar a sequência de caracteres recebida é apagado. A lista de todos os comandos utilizados encontra-se disponível nas Tabelas 8.2, 8.3, 8.4 e 8.5.

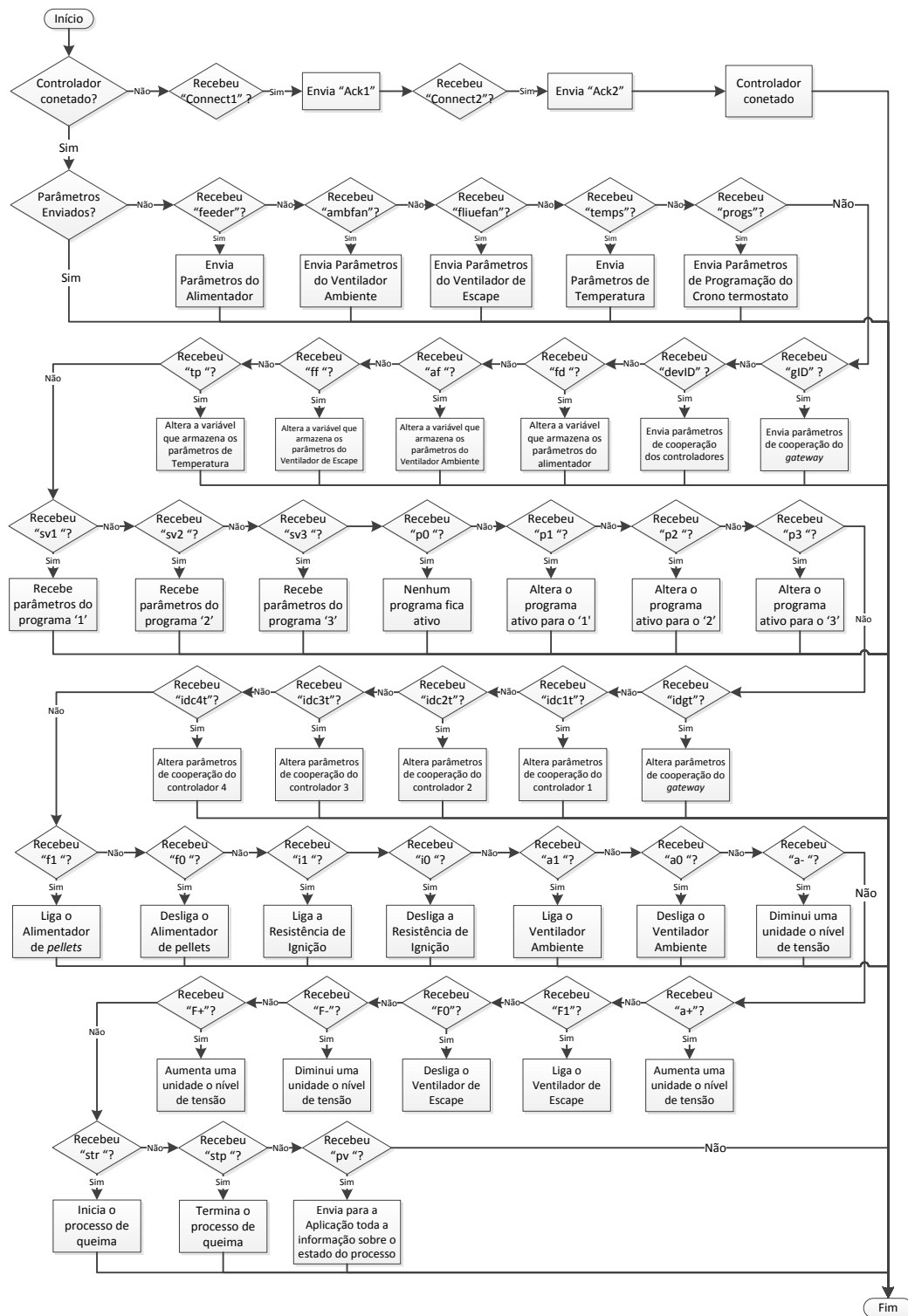


Figura 5.13 - Fluxograma da RSI para a recepção de caracteres

Em JAVA, a implementação da *API* de comunicações está acessível a partir da biblioteca *"RXTXcomm"* e nela estão contidas rotinas básicas de gestão desta interface de comunicação. Contudo, não existe uma rotina específica para realizar ações de acesso à porta série para leitura ou escrita, como na linguagem C.

Para ser possível a leitura ou envio de dados para a porta série é necessário obter previamente informação sobre as portas que estão a ser utilizadas pelo *PC*. Neste caso, a porta deve ser do tipo série e o seu nome corresponder ao elemento da lista especificado pelo instalador (COM4, por exemplo) na aplicação.

A partir daqui é invocado um construtor, que é responsável pela abertura da respetiva porta, pela definição dos parâmetros de comunicação (*baud rate*, *data bits*, *stop bits* e *paridade*) e pela restante configuração necessária para a leitura de dados da porta série (Figura 5.14):

- Criação de um *stream* de dados do tipo *InputStream*, para a receção;
- Criação de um “*Listener*” para gerar um evento sempre que uma mudança no conteúdo da porta série for detetada;
- Definição dos parâmetros de configuração da porta;
- Criação e inicialização de uma *thread*.

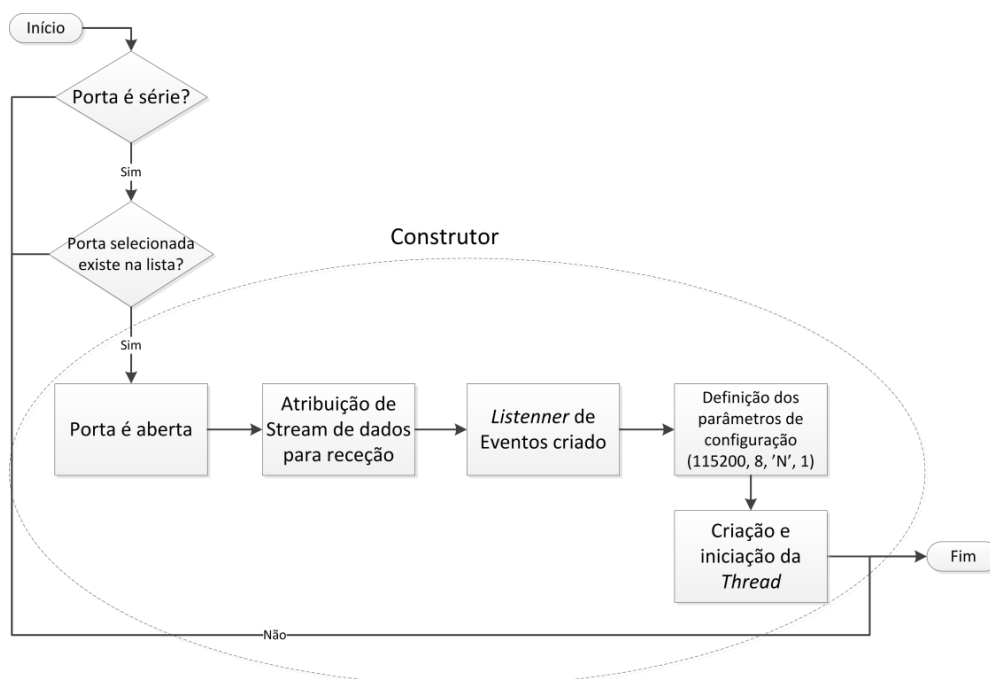


Figura 5.14 - Configuração para a leitura de caracteres da porta série

Dentro da rotina que é gerada a cada novo evento ocorrido na porta série, é necessário depositar os dados recebidos para um *array* de caracteres (Figura 5.15) que possa ser acedido no programa principal.

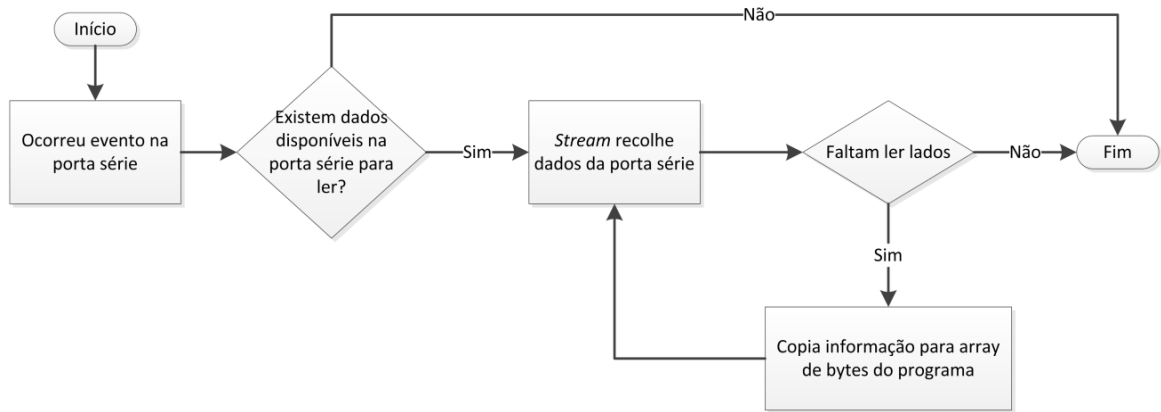


Figura 5.15 - Fluxograma da leitura de dados da porta série

A escrita é mais simples de implementar, uma vez que não utiliza a *thread* nem a geração de eventos assíncronos. No programa principal, sempre que se pretenda escrever sobre a porta série basta atribuir ao objeto de *stream* de dados para transmissão do tipo *outputStream*, o conteúdo da *string* que se pretende enviar e utilizar a rotina “*write*” associada a esta classe (Figura 5.16) para enviar a informação.

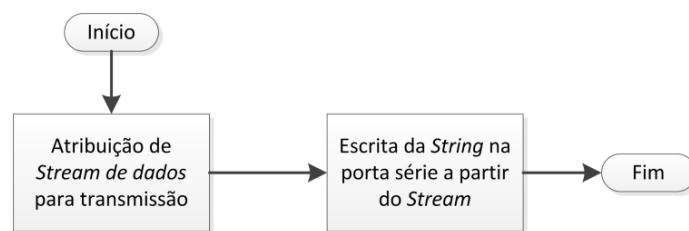


Figura 5.16 - Fluxograma da escrita de dados na porta série

5.2.2 - ESTRUTURA DA APLICAÇÃO

A aplicação estrutura-se por abas, sendo que cada uma delas está destinada a uma ou mais operações específicas.

- Conexão

A aba de conexão é responsável pelo estabelecimento da ligação entre o computador e o controlador, assim como do acompanhamento e configuração dos parâmetros de cooperação dos restantes controladores ligados ao barramento.

A conexão deve ser a primeira atividade a ser executada, para garantir que se estabeleceu uma correta comunicação entre ambos. Assim sendo, inicialmente quando a aplicação é executada, deve encontrar-se desconectada (Figura 5.17)

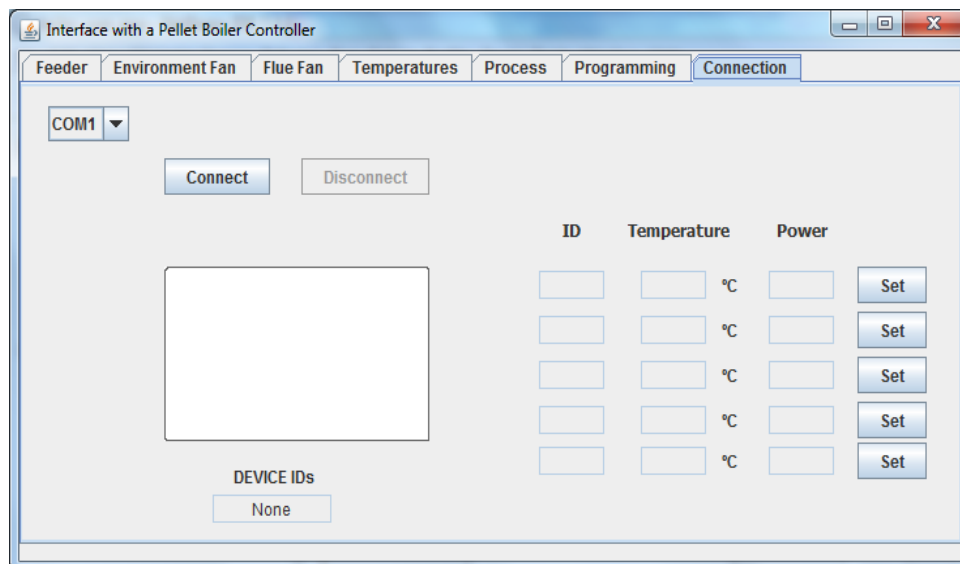


Figura 5.17 - Aplicação inicialmente desconectada

O botão “*Connect*” deve ser clicado para estabelecimento da ligação, iniciando-se uma troca de mensagens segundo a Figura 5.18. Para essa troca de mensagens se poder efetivar com sucesso são necessárias algumas validações, como por exemplo se a porta selecionada da lista coincide com a que está ligada ao controlador, entre outras (Figura 5.19). Recebendo a segunda mensagem de confirmação, “Ack2”, a aplicação considera ter o controlador conectado e pede que lhe envie todos os parâmetros do sistema, atualizando os campos de dados respectivos.

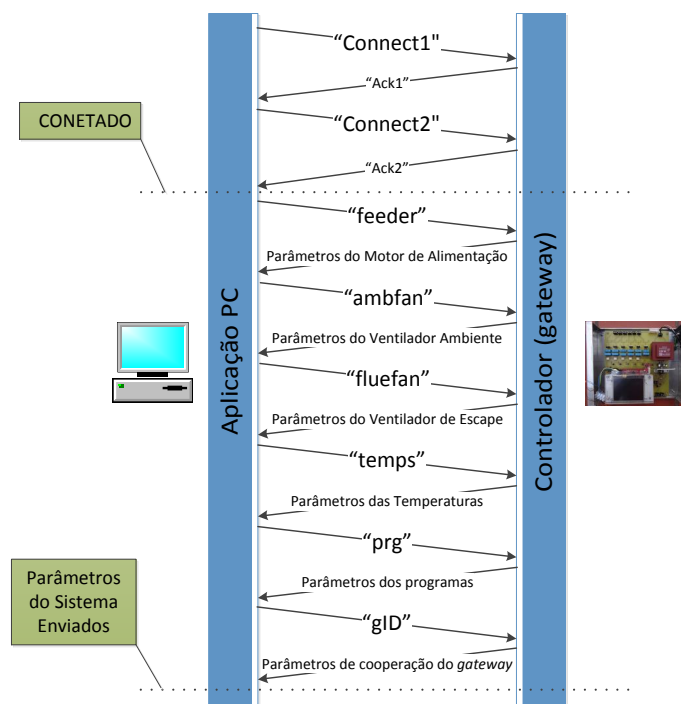


Figura 5.18 - Troca de mensagens despoletada pelo botão “*Connect*”

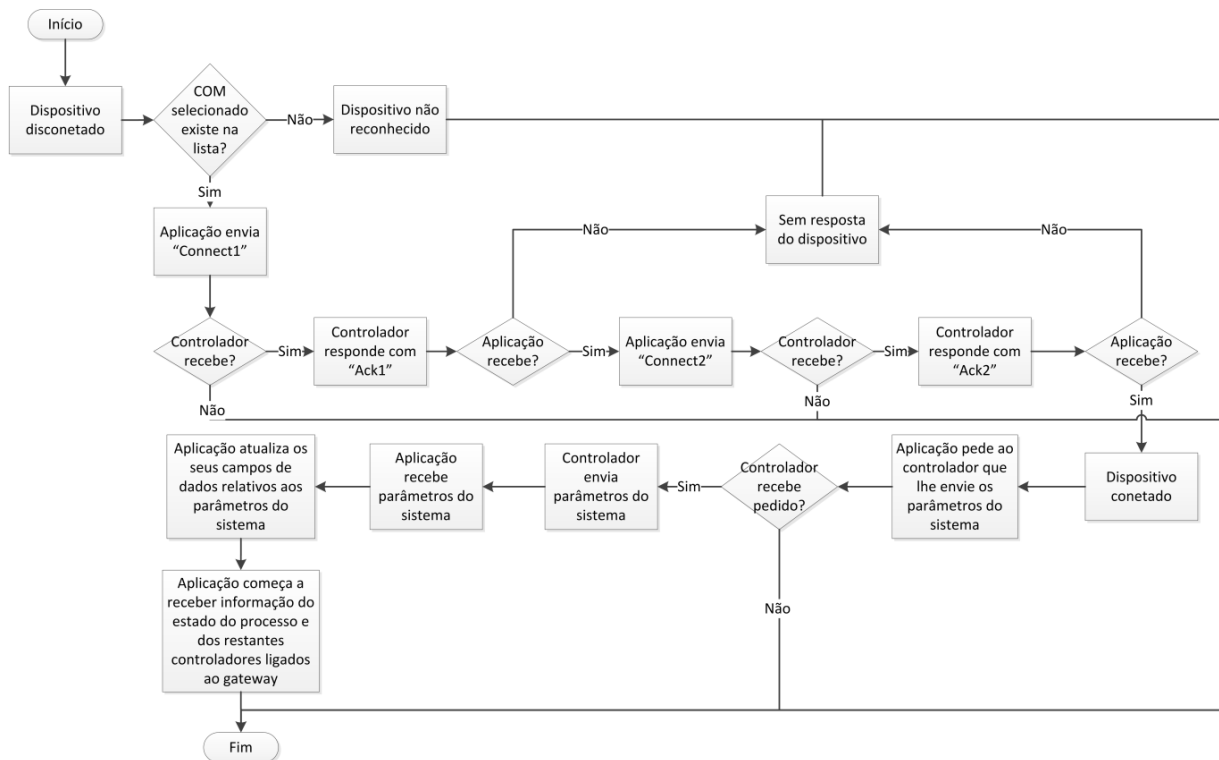


Figura 5.19 - Fluxograma estabelecido pelo botão “Connect”

- Configuração dos parâmetros do sistema

Os mecanismos de configuração dos parâmetros do sistema anteriormente descritos são descritos nesta secção. A Figura 5.20 diz respeito à aba do alimentador de *pellets* onde podem ser visualizados os campos de dados relativos aos seus parâmetros.

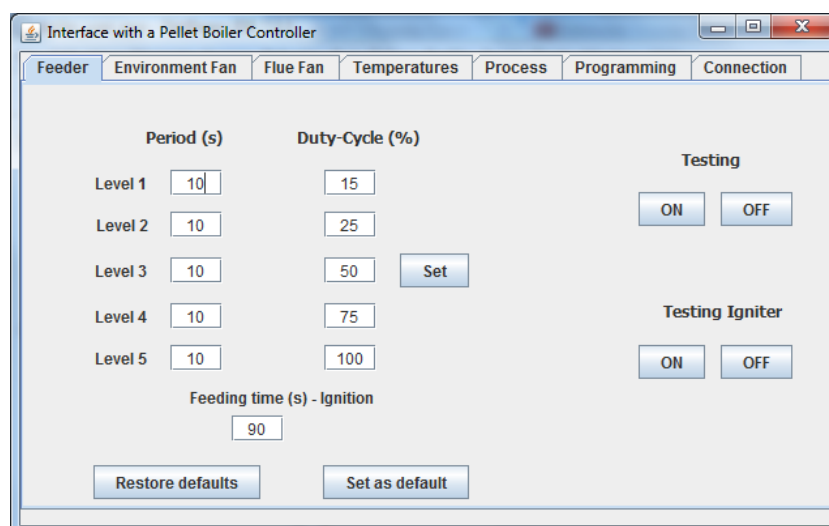


Figura 5.20 - Aba para alteração de parâmetros e teste do MA e RI

Em caso de alteração de um dos parâmetros mencionados em cima, podem ser realizadas três ações. Uma delas é despoletada pelo botão “Set” que envia a

informação para a variável respetiva do programa que corre no controlador. A segunda é despoletada pelo botão “*Set default*” que, além de fazer o mesmo que o anterior, pede ainda que a nova alteração seja guardada na memória do controlador, como predefinição. Por último, no caso de o instalador ter alterado os valores predefinidos por outros sem os ter predefinido novamente e pretender recuperá-los, em vez de os estar a introduzir todos novamente, basta que clique no botão “*Restore defaults*”. Voltando à aba de conexão (Figura 5.17), podem ser observados nos campos de texto mais à direita os parâmetros de identificação, as temperaturas a que entram em funcionamento e o nível de queima dos controladores presentes no barramento. Ocorrida uma alteração de um destes parâmetros, é enviada a nova informação, através do botão “*Set*”, para o *gateway*. Por sua vez, este encarregar-se-á de enviá-la ao(s) respetivo(s) controladores, que guardarão a alteração solicitada.

- Teste aos atuadores

Esta funcionalidade da aplicação destina-se, durante a instalação e/ou reparação do aparelho, a certificar que os atuadores estão a trabalhar como pretendido, facilitando a deteção de eventuais avarias.

O teste ao motor de alimentação e à resistência de ignição (Figura 5.20) apenas é feito com um botão para ligar, “*ON*”, e outro para desligar, “*OFF*”. Os testes aos ventiladores, além de contemplarem estas duas ações, possuem também dois botões para regulação do nível de tensão a elas aplicada, “*+*” e “*-*”, e consequentemente da sua velocidade de rotação. A Figura 5.21 exemplifica o teste que pode ser realizado ao ventilador ambiente.

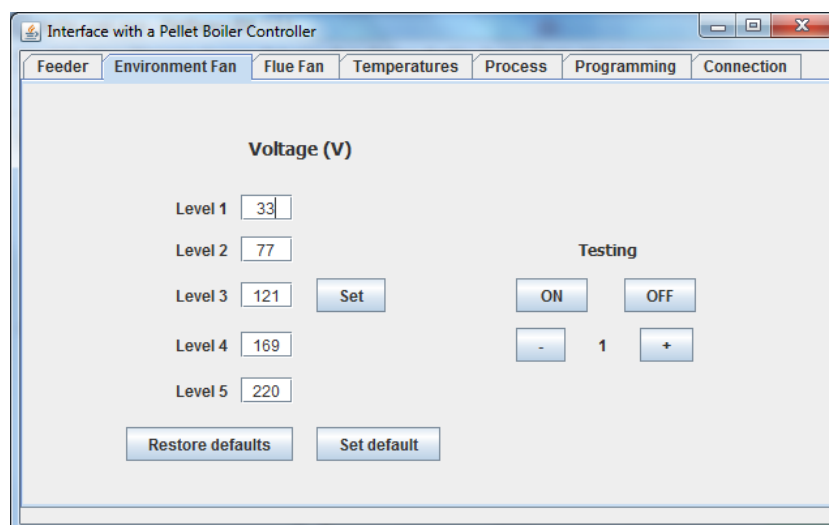


Figura 5.21 - Aba para alteração de parâmetros e teste do VA

- Monitorização do estado do processo e dos controladores em cooperação

Uma vez que o instalador/operador de manutenção do sistema deve ter acesso ao maior número de informação possível, foi criada a aba da Figura 5.22 com o objetivo de verificar o correto funcionamento da máquina de estados, através da leitura do estado dos atuadores, da fase do processo e das temperaturas medidas.

Depois de estabelecida a conexão com o controlador, nova informação referente ao estado do processo é recebida periodicamente pela aplicação, atualizando assim os campos de texto de 2 em 2 segundos. Este período foi escolhido tendo em conta que mudanças do estado do processo devem ser observadas com alguma brevidade.

Inicialmente, o processo encontra-se no estado “Inativo”, logo os seus atuadores estarão todos desligados e a temperatura dos gases de escape igual à temperatura da placa. Para iniciar o processo é necessário um clique no botão “START” ficando entregue todo o controlo à máquina de estados. Para terminá-lo basta um clique no botão “STOP”, voltando ao estado inicial, depois da extinção da chama.

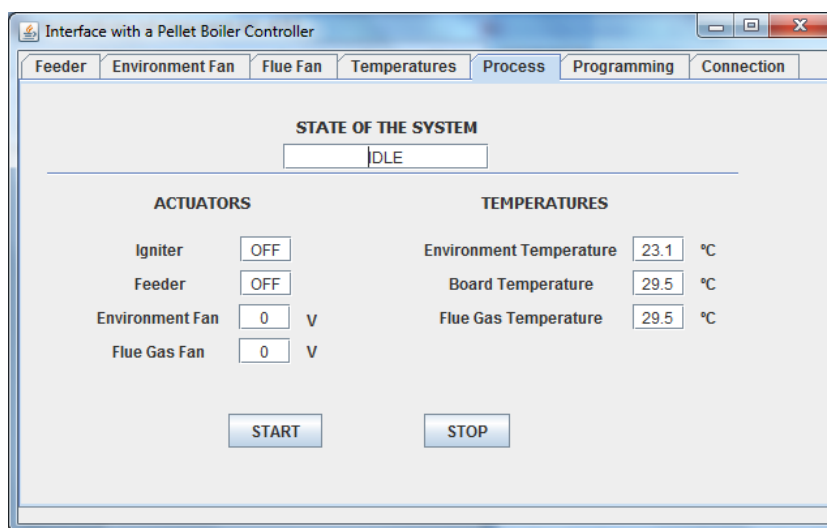


Figura 5.22 - Ecrã de monitorização do processo de aquecimento

Voltando à aba de conexão, onde é igualmente realizada a monitorização dos controladores de cooperação, além do *gateway*, podem ser observados os seus números identificadores no campo de dados, em tempo real (Figura 5.17). Esta lista é atualizada continuamente com o intuito da deteção de novo dispositivo ser feita de forma rápida por parte do instalador/operador de manutenção, assim como a ocorrência de falha, avaria ou ainda de remoção.

Estas duas monitorizações são conseguidas, novamente, através de uma troca de mensagens realizada periodicamente de 2 em 2 segundos (Figura 5.23).

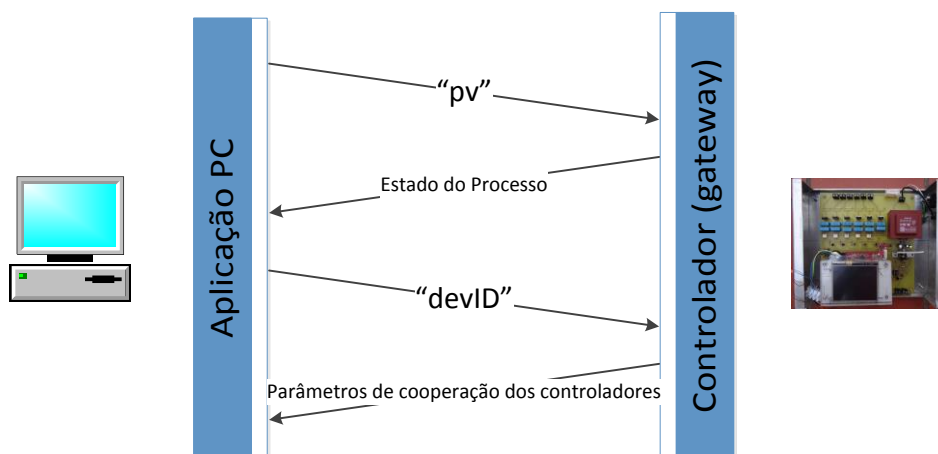


Figura 5.23 - Troca de mensagens de monitorização entre o *PC* e o *gateway*

Na Tabela 5.6, encontram-se condensadas as abas por tipo de tarefa.

Tarefa	Abas	Funcionalidades
Conexão	" <i>Connection</i> "	Estabelecimento da ligação entre o controlador e a aplicação <i>PC</i>
Configuração dos parâmetros do sistema	" <i>Feeder</i> "/ " <i>Environment Fan</i> "/ " <i>Flue Fan</i> "	Configuração dos parâmetros associados aos cinco níveis da fase de trabalho e à fase de ignição. As alterações efetuadas podem ordenar a salvaguarda na variável do programa e/ou na memória do controlador.
	" <i>Temperatures</i> "	Configuração dos parâmetros associados às temperaturas.
	" <i>Programming</i> "	Configuração dos parâmetros associados ao crono termostato.
Teste aos atuadores	" <i>Feeder</i> "/ " <i>Environment Fan</i> "/ " <i>Flue Fan</i> "	Ligar e desligar cada um dos atuadores; Regular a velocidade dos ventiladores entre cinco níveis.
Monitorização do estado do processo e dos controladores em cooperação	" <i>Connection</i> " " <i>Process</i> "	Iniciar e terminar o processo de aquecimento; Visualização das temperaturas medidas pelos sensores, do estado dos atuadores e do próprio processo; Visualização dos controladores que se encontram ligados ao barramento em tempo real.

Tabela 5.6 - Funcionalidades disponíveis na aplicação *PC* por tarefa

Capítulo 6 TESTES DE VALIDAÇÃO

Os testes realizados destinaram-se a validar o comportamento dos sensores, atuadores e respetivas interfaces de acondicionamento, assim como a máquina de estados.

6.1 TESTES AOS SENSORES

Com o objetivo de encontrar um termo de comparação as experiências realizadas sobre os sensores foram realizadas sempre com o auxílio de um termómetro FLUKE.

6.1.1 - NTC

Num primeiro teste foram colocados o termómetro FLUKE e a NTC a medir a temperatura ambiente de uma sala de estar com portas e janelas abertas para forçar o seu valor a ser o mais constante possível durante a experiência, dada a elevada massa de ar em contínua circulação. Durante cerca de 25 minutos foram recolhidas amostras separadas por 30 segundos (Gráfico 6.1). Houve o cuidado de colocar ambos os sensores à mesma altura e à mesma distância de uma das paredes para uniformizar a medida.

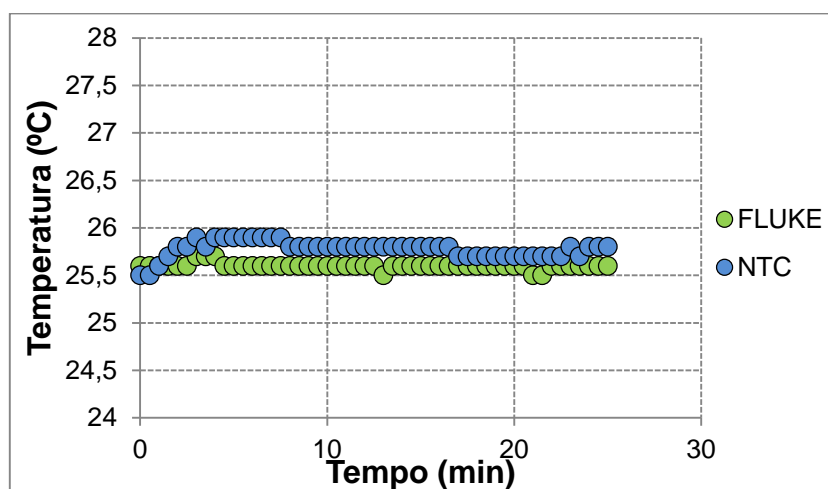


Gráfico 6.1 - Medição da temperatura ambiente

O valor máximo atingido pelo termómetro FLUKE foi de 25,7°C, enquanto com a NTC de 25,9°C. O valor mínimo situou-se nos 25,5°C em ambos os sensores.

O erro máximo verificado foi de -0,3°C (Gráfico 6.2), o que em termos práticos tem pouco significado, uma vez que o utilizador não é capaz de se aperceber desta

diferença no que diz respeito à temperatura ambiente. Pode-se concluir que a precisão e resolução na medição da temperatura da NTC cumprem os requisitos.

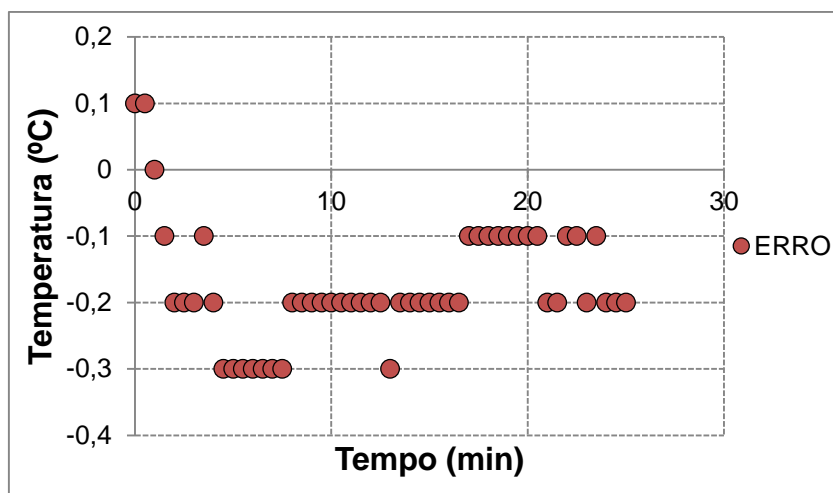


Gráfico 6.2 - Erro associado à medição da temperatura ambiente

Num segundo teste arrefeceu-se água de uma pequena bacia com cerca de 1 litro de capacidade, no congelador, e retirou-se a mesma deixando-a repousar e iniciando de imediato a medição da sua temperatura.

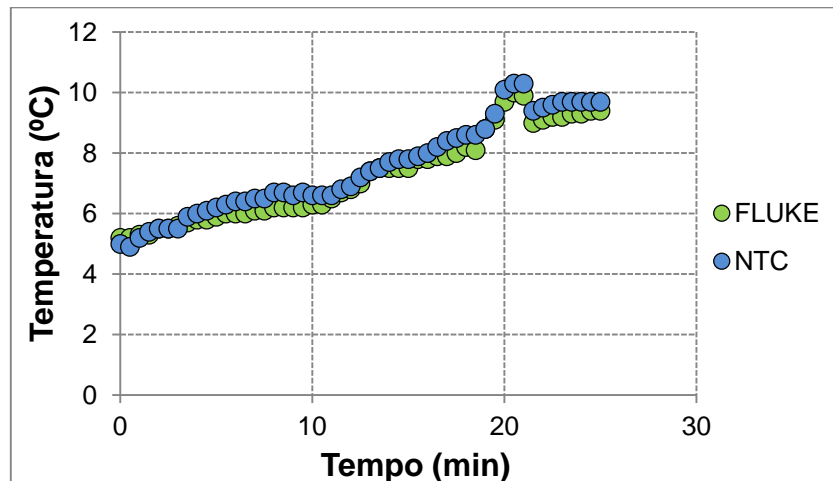


Gráfico 6.3 - Medição da temperatura da água

Verifica-se que, inicialmente, a temperatura da água era de cerca de 5°C e ao final de 25 minutos esta aumentou até aos 10°C (Gráfico 6.3).

O erro absoluto, associado à diferença entre a temperatura medida pelo termómetro FLUKE e a NTC, teve o seu máximo em 0,5°C (Gráfico 6.4). Algumas oscilações entre as duas medições podem dever-se ao facto de durante a experiência terem sido colocadas as mãos várias vezes em contacto com a bacia, para fazer aquecer mais rapidamente a água no seu interior.

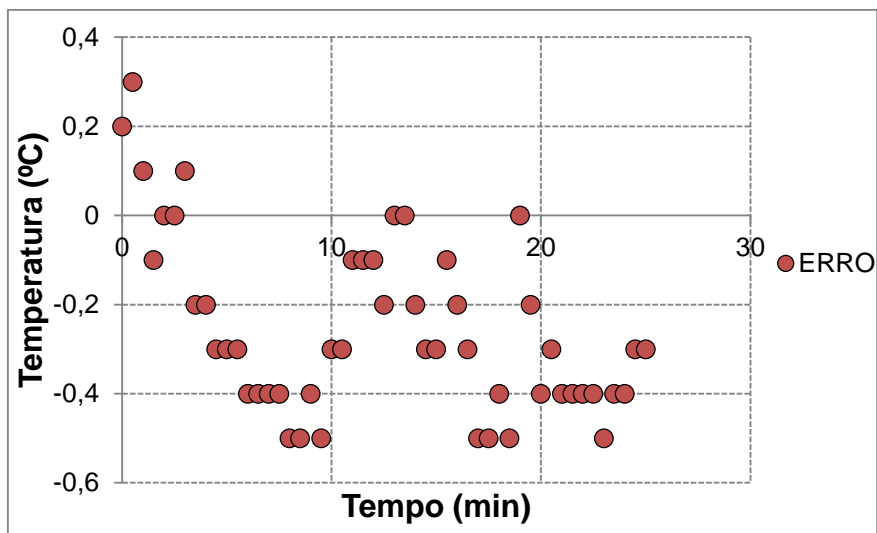


Gráfico 6.4 - Erro associado à medição da temperatura da água

Novamente, a dispersão de pontos relativos ao erro se dá predominantemente em ordenadas negativas, o que significa que se pode proceder a um pequeno ajuste na medição da temperatura da NTC, nomeadamente reduzir um pouco o seu *offset*.

6.1.2 - TERMOPAR K

Com o auxílio de uma torradeira foram colocados o termómetro FLUKE e o Termopar suspensos, mas próximos da superfície metálica aquecida. Inicialmente deixou-se que a torradeira atingisse a sua temperatura máxima para, posteriormente, desligá-la da corrente e iniciar-se a medição do ar quente por ela irradiado.

Este teste teve a duração de 10 minutos, com um intervalo de 20 segundos entre cada amostra (Gráfico 6.5).

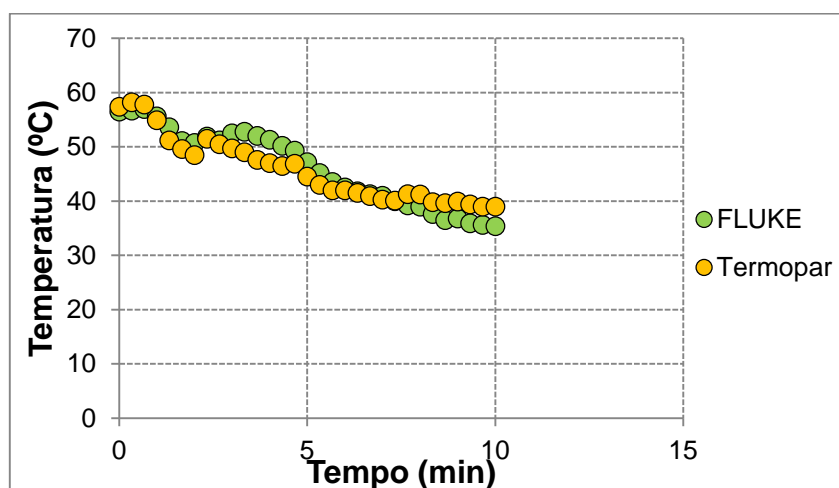


Gráfico 6.5 - Medição da temperatura do ar quente

Observam-se em vários períodos de tempo que a temperatura medida pelo termopar é ligeiramente superior à do FLUKE e vice-versa.

O erro máximo obtido situou-se em 4.4°C (Gráfico 6.6), enquanto a maior diferença de erro entre duas amostras foi de 2°C, o que na aplicação em causa (medição da temperatura dos gases de escape) não é crítico, uma vez que não lhe é exigida uma precisão inferior a esse valor.

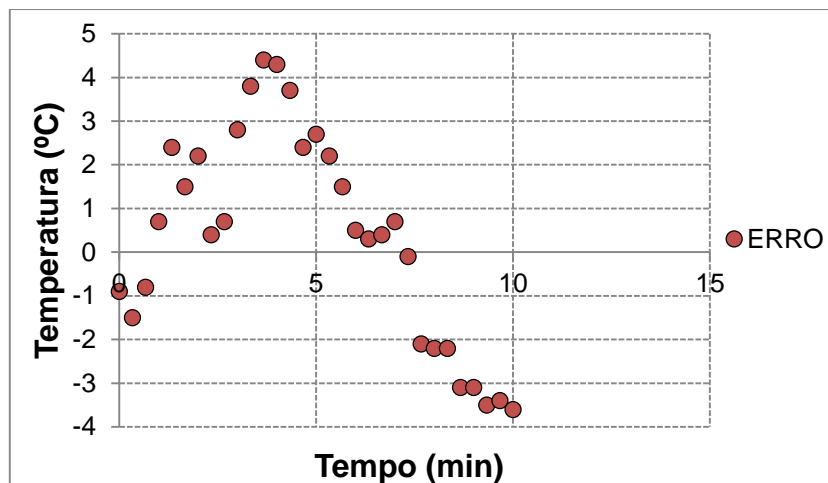


Gráfico 6.6 - Erro associado à medição da temperatura do ar quente

6.1.3 - LM335

O teste realizado com o sensor de temperatura destinado à compensação da junção fria do termopar foi realizado durante 22 minutos e iniciou-se logo após ter sido ligado o controlador, com um intervalo de 30 segundos entre cada amostra.

Os Gráficos 6.7 e 6.8 descrevem respetivamente a evolução da temperatura da placa, medida através do termómetro FLUKE e LM335 e o erro associado.

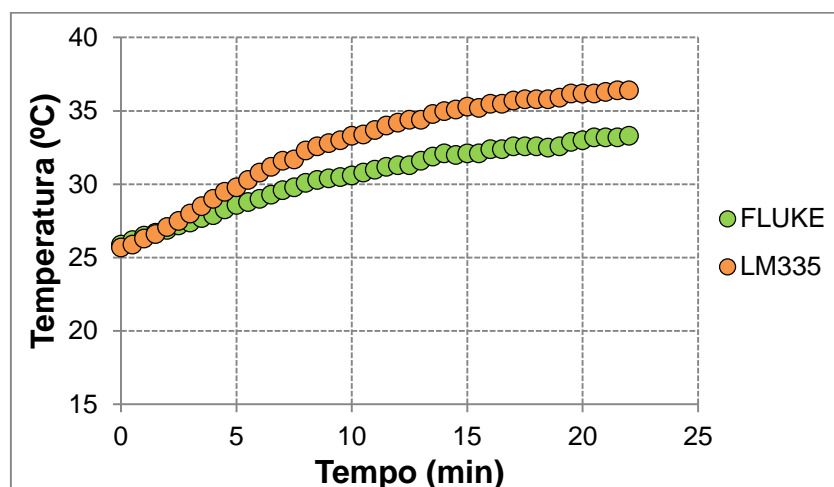


Gráfico 6.7 - Medição da temperatura da placa

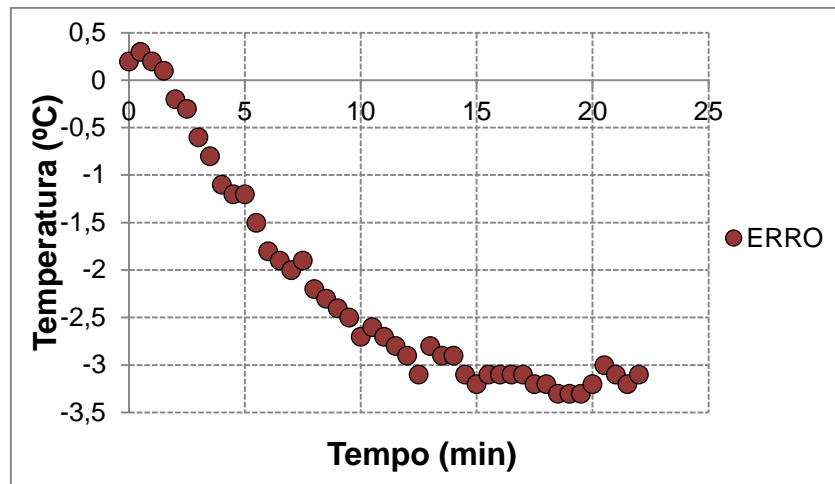


Gráfico 6.8 - Erro associado à medição da temperatura da placa

Verificou-se que a temperatura da placa sobe continuamente, o que se justifica pela elevada dissipação de calor provocada pelo regulador de tensão LM7805, que fornece a alimentação necessária a grande parte dos componentes. A calibração deste sensor teve em conta a temperatura ambiente inicial medida pelo termómetro FLUKE e foi realizada por *software*.

Visualizando os dois gráficos conclui-se que o erro vai aumentando (em módulo) à medida que a temperatura da placa também sobe. Isto deve-se ao facto da junção do termopar do FLUKE ter sido encostado ao invólucro plástico do sensor e este não attingir a temperatura das pistas de sinal que o atravessam (a saída do LM335 está ligado aos 5V através de apenas uma resistência logo, encontra-se mais quente). Assim, a medida não foi realizada no mesmo local em ambos os sensores. Daí o erro máximo ter chegado aos 3,3°C. Apesar disso, a resolução e precisão na medição da temperatura para o LM335 cumprem os requisitos (0,1°C).

6.2 ATUADORES

As medições de corrente e tensão foram efetuadas nesta secção com o auxílio de um multímetro FLUKE de *true rms*, uma vez que as formas de onda geradas devidas à modulação do ciclo (ângulo de disparo do *TRIAC*), não são sinusoidais, o que utilizando o método do valor médio se estariam a cometer erros grosseiros.

6.2.1 - TESTE EM VAZIO

O primeiro teste foi realizado sem qualquer atuador ligado, com o objetivo de se medir a corrente que circula no lado primário do transformador (Figura 6.1). O valor em repouso, ou seja, apenas com a placa ligada foi de 0,061 A, enquanto o valor máximo

foi de 0,075 A e verificou-se quando o *bip* do altifalante do módulo VM800C era ativado, através do toque num qualquer “botão” do ecrã do *LCD* de forma repetitiva.

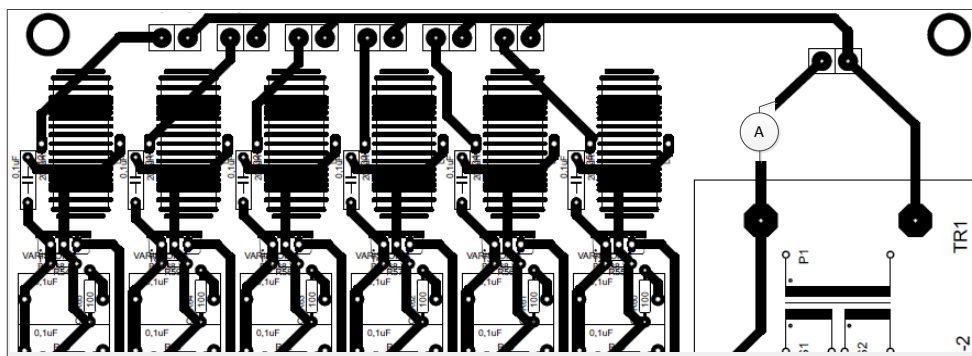


Figura 6.1 – Montagem da experiência em vazio

6.2.2 - TESTE COM CARGAS INDIVIDUAIS

Nesta secção encontram-se as medidas de corrente alternada na linha que alimenta as cargas (Fase ou Neutro da rede) e a tensão eficaz alternada aos terminais do atuador respetivo, colocado em funcionamento com todos os restantes (Figura 6.2).

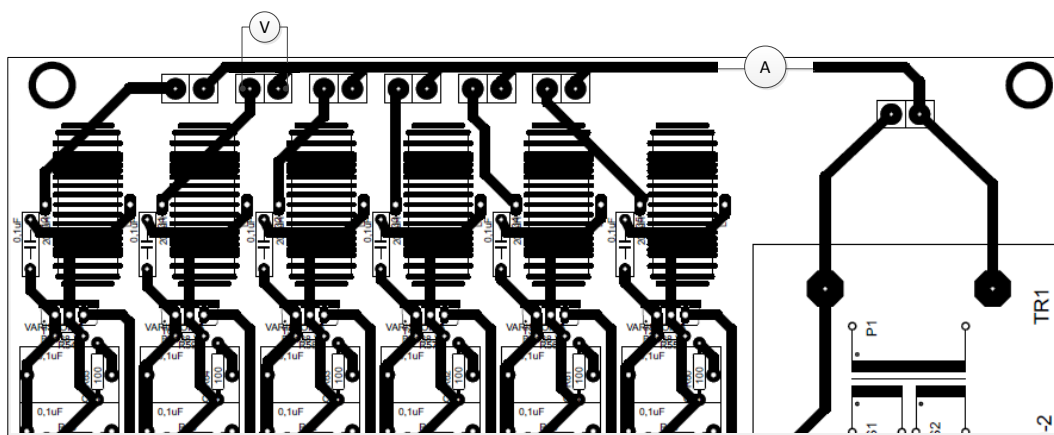


Figura 6.2 – Montagem da experiência com a RI ligada

Os testes realizados com a resistência de ignição (Tabela 6.1) mostram que, logo após esta entrar em funcionamento, se encontra a ser percorrida por uma corrente elevada mas que rapidamente decresce até se situar nos 0,719 A. A corrente consumida pelo motor de alimentação é cerca de 0,336 A durante o período em que se encontra ligado (Tabela 6.2). Relativamente aos ventiladores, mediram-se as mesmas grandezas elétricas, mas dividindo-as nos seus cinco níveis de funcionamento (Tabelas 6.3 e 6.4). Comparando ambos, verifica-se que o ventilador de escape consome sempre mais corrente em qualquer um dos níveis, assim como a tensão aos seus terminais é superior, excetuando nos níveis um e dois.

Resistência de ignição		
Fase	Corrente (A)	Tensão (V)
Inicial	2,339	236,2
Final	0,719	238,1

Tabela 6.1 - Corrente e tensão na RI

Motor de alimentação	
Corrente (A)	Tensão (V)
0,253	239,7

Tabela 6.2 - Corrente e tensão no MA

Ventilador ambiente		
Nível	Corrente (A)	Tensão (V)
1	0,168	169,6
2	0,211	184,5
3	0,252	197,9
4	0,310	212,7
5	0,389	227,5

Tabela 6.3 - Corrente e tensão no VA

Ventilador de escape		
Nível	Corrente (A)	Tensão (V)
1	0,183	157,8
2	0,252	179,3
3	0,320	197,2
4	0,420	215,5
5	0,478	233

Tabela 6.4 - Corrente e tensão no VE

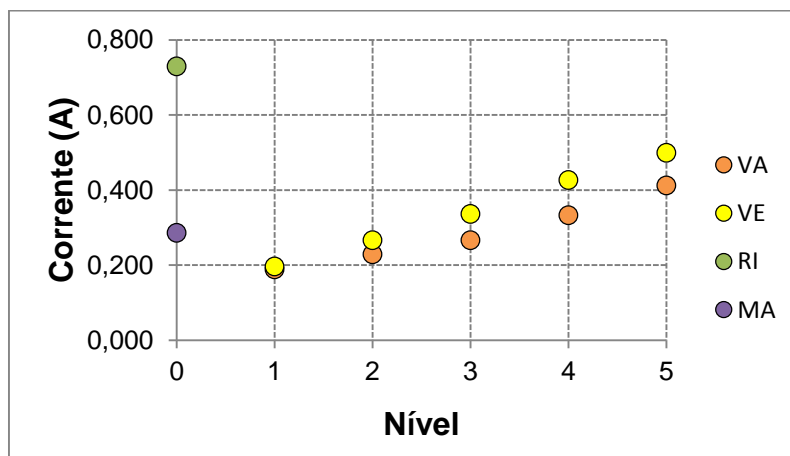


Gráfico 6.9 - Correntes medidas no teste individual aos atuadores

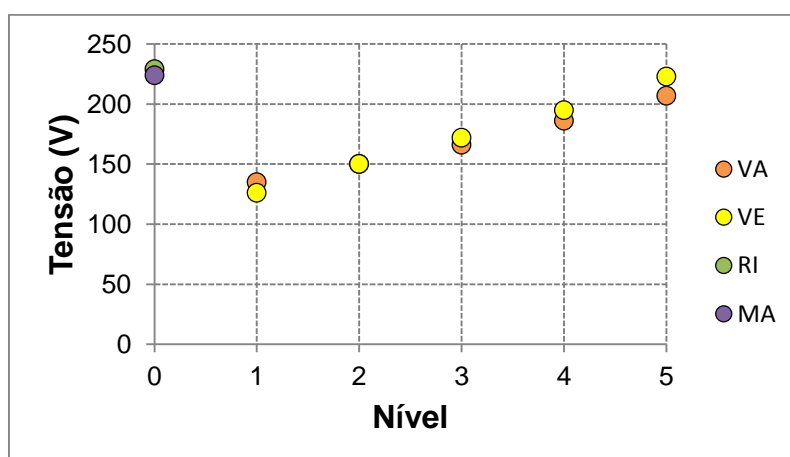


Gráfico 6.10 - Tensões medidas no teste individual aos atuadores

Os Gráficos 6.9 e 6.10 traduzem visualmente os resultados apresentados.

6.2.3 - TESTE COM CARGAS SIMULTÂNEAS

Nesta secção procurou-se observar a corrente total consumida por vários atuadores ligados ao mesmo tempo e a tensão medida aos terminais do ventilador de escape.

Inicialmente ligaram-se ambos os ventiladores e mediram-se todas as combinações entre níveis de funcionamento (Tabela 6.5). A montagem segue o mesmo esquema dos testes anteriores.

Ventilador de Escape										
Nível	1		2		3		4		5	
	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)
1	0,416	157,5	0,485	178,6	0,549	196,3	0,644	215,1	0,7	230,8
2	0,458	157,6	0,526	178,6	0,592	196,1	0,688	214,9	0,742	230,6
3	0,496	157,4	0,566	178,6	0,634	196,1	0,728	214,8	0,781	230,6
4	0,552	157,6	0,622	178,1	0,689	196	0,783	214,8	0,842	230,5
5	0,630	157,5	0,701	178,1	0,767	195,9	0,861	214,6	0,921	230,7

Tabela 6.5 - Medições com apenas os ventiladores ligados

Conclui-se, como esperado, que quanto maior a soma do nível de funcionamento de ambas, maior a corrente consumida. Contudo, a tensão medida aos terminais do ventilador de escape praticamente não sofreu alterações, variando o nível de funcionamento do ventilador ambiente, ou seja, não é influenciada por outra carga ligada a uma saída diferente.

Na experiência seguinte ligou-se o motor de alimentação, continuando com os ventiladores ligados. Incrementos ao nível da corrente total consumida são notórios (Tabela 6.6).

Ventilador de Escape										
Nível	1		2		3		4		5	
	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)
1	0,73	155,6	0,793	176,9	0,861	195,1	0,948	212,9	1,006	230,2
2	0,769	155,4	0,837	176,9	0,908	195,1	0,991	212,9	1,048	229,9
3	0,811	155,4	0,875	176,9	0,947	194,8	1,032	212,8	1,089	229,8
4	0,863	155,7	0,932	177	1,002	194,9	1,082	212,6	1,147	229,9
5	0,94	155,6	1,006	177	1,074	194,7	1,164	212,7	1,225	229,9

Tabela 6.6 - Medições com os ventiladores e o motor de alimentação ligados

No último teste, ligou-se ainda, a resistência de ignição e fizeram-se as respetivas medições. O valor máximo de corrente consumida pelos 4 atuadores situou-se em

1,719 A (Tabela 6.7), curiosamente um valor inferior ao consumido pela resistência de ignição na sua fase inicial.

Ventilador de Escape										
Nível	1		2		3		4		5	
	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)	Corrente (A)	Tensão (V)
1	1,239	155,2	1,297	176,2	1,357	196,6	1,451	215,5	1,514	232,1
2	1,277	155,4	1,334	175,9	1,391	196,7	1,493	215,5	1,555	231,9
3	1,317	155,5	1,376	176	1,427	196,6	1,526	215,2	1,594	232,2
4	1,368	155,2	1,432	176,1	1,482	196,4	1,585	215,2	1,641	232
5	1,443	155,1	1,509	175,9	1,561	196,5	1,66	215,2	1,719	231,9

Tabela 6.7 - Medições com todos os atuadores ligados

Nas três experiências o valor de tensão aos terminais do ventilador escape, permaneceu praticamente constante em qualquer um dos níveis, o que permite concluir que o canal é robusto contra interferências externas por parte de outros atuadores.

Fazendo uma comparação entre a soma das correntes consumidas individualmente, por cada um dos atuadores relativamente aos valores quando as cargas se encontram ligadas em simultâneo, verifica-se que a primeira excede a segunda. O Gráfico 6.11 mostra os resultados de ambas as partes, para um nível fixo do ventilador ambiente de 3, fazendo variar o do ventilador de escape.

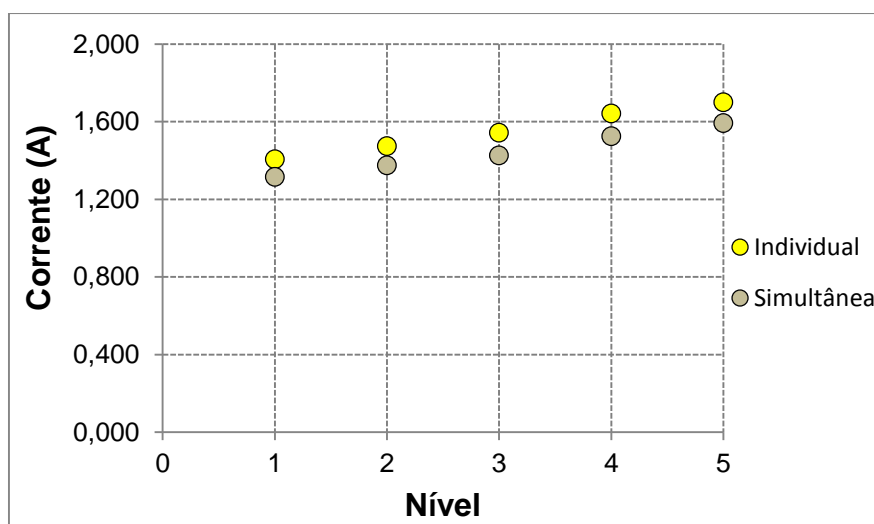


Gráfico 6.11 - Corrente total consumida a partir das contribuições individuais vs simultânea

6.3 COMUNICAÇÃO ENTRE CONTROLADORES

Nesta secção encontram-se os testes feitos à comunicação entre controladores.

Antes de ser dado início à experiência, programaram-se os dois microcontroladores da parte inferior da Figura 6.3 para enviarem os seus três parâmetros de 2 em 2 segundos através de uma mensagem *CAN*, sem sincronismo entre ambos. O controlador principal foi configurado para receber mensagens de cada um deles e atualizar com a mesma frequência o campo de dados da aplicação que corre sobre o *PC*.

Inicialmente, foi estabelecida a comunicação entre o *gateway* e o *PC* com os microcontroladores desligados e por isso apenas este se encontra presente no barramento (Figura 6.4)

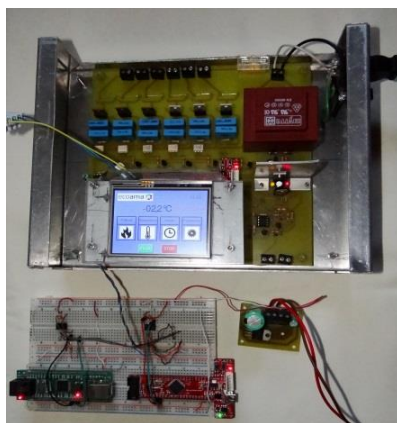


Figura 6.3 - Montagem de teste do barramento *CAN*

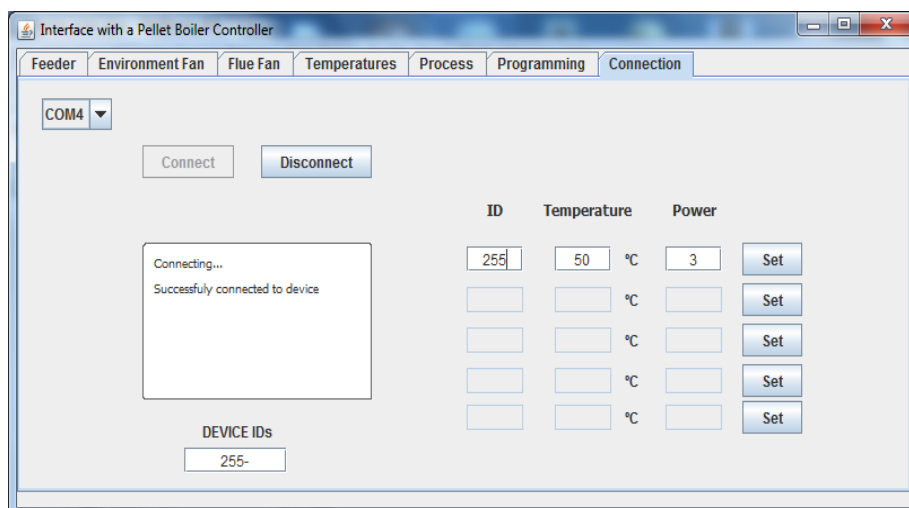


Figura 6.4 - Apenas o *gateway* ligado ao barramento *CAN*

De seguida, foram ligados os dois microcontroladores em simultâneo. O *gateway* recebeu essa informação e ordenou-a em termos de número identificador enviando imediatamente para a aplicação, que atualizou os seus campos de dados (Figura 6.5).

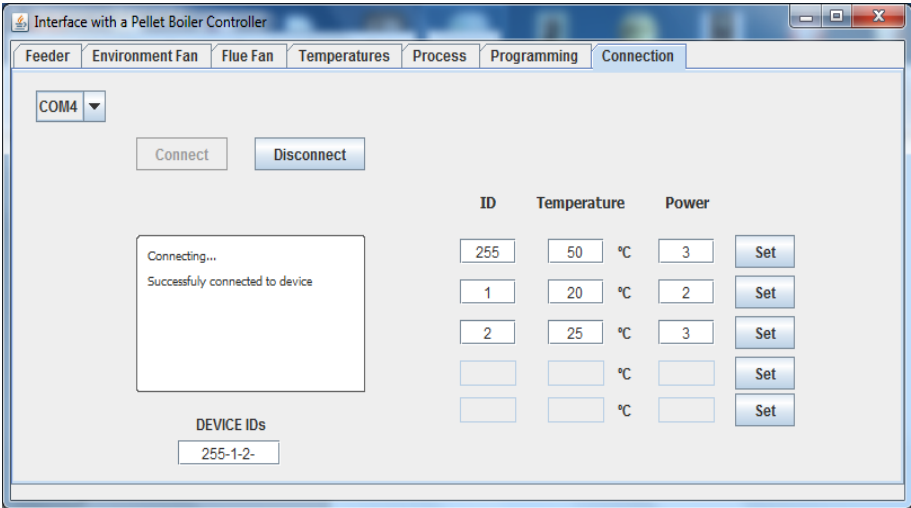


Figura 6.5 - Dois controladores ligados ao barramento

O teste seguinte foi o de desligar o microcontrolador com o número identificador 2, cuja alteração se encontrou rapidamente visível na Figura 6.6.

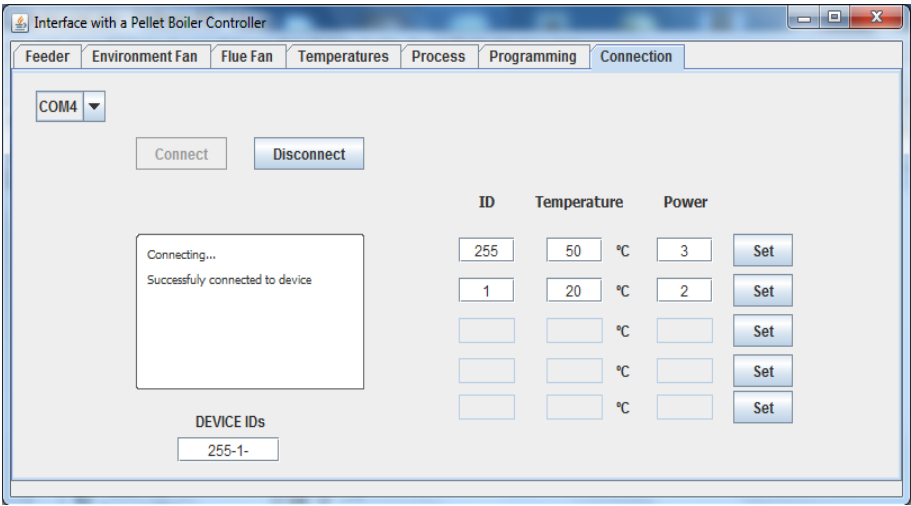


Figura 6.6 - Remoção do controlador 2

Por último alterou-se o número identificador do controlador 1, assim como a temperatura de funcionamento e o poder de queima, desligando-se logo após o *gateway*. Assim, que a ligação foi restabelecida, as alterações solicitadas foram observadas no campo de dados destinado para o efeito (Figura 6.7).

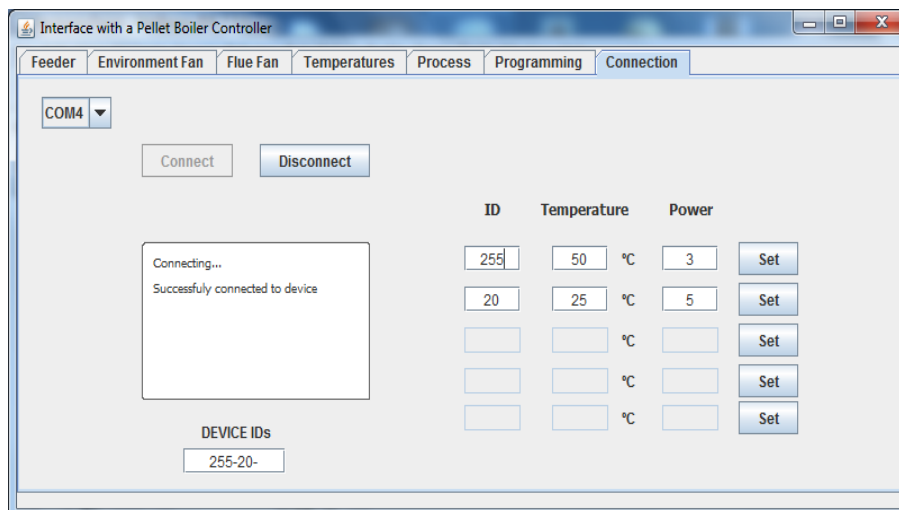


Figura 6.7 - Alteração dos parâmetros do controlador 1

Conclui-se que a comunicação *CAN* estabelecida entre controladores se dá com sucesso e funciona como pretendido.

6.4 CONTROLADOR EM AMBIENTE REAL

Com o objetivo de validar a máquina de estados, as interfaces de atuação e sensores, assim como a comunicação série, o controlador desenvolvido foi testado na empresa ECOAMA, mais concretamente no recuperador de calor Héstia (Figura 6.8 – aparelho do lado direito com tons de vermelho).



Figura 6.8 - Controlador em teste nas instalações da ECOAMA

Depois de conectados todos os atuadores e sensores do aparelho à placa do controlador, ligou-se ao *PC* (via RS232 com conversor USB) que corria um terminal de dados. Inicialmente, o estado do sistema é “Inativo”, tendo-se medido uma temperatura ambiente de 24,6°C. Antes de se ligar o processo de aquecimento,

selecionou-se a temperatura desejada para 30°C, de forma a colocar o nível de queima no máximo (nível 5). Pressionado o botão “START” do *LCD*, o processo iniciou-se e a máquina de estados transitou para o estado de “Ignição” como previsto, começando a visualizar-se os valores de temperatura medidos no terminal de dados do *PC*, de 5 em 5 segundos.

No Gráfico 6.12 pode ser vista a evolução da temperatura ambiente, dos gases de escape e da placa durante os cerca de 27 minutos de teste.

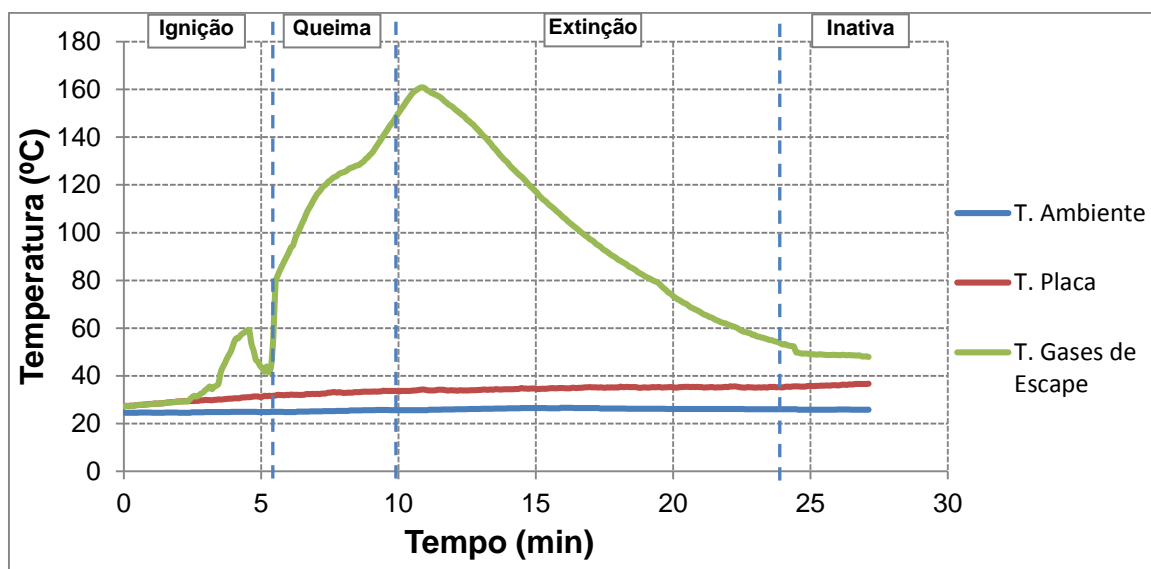


Gráfico 6.12 - Evolução das temperaturas ao longo do processo

Passados pouco menos de 5 minutos desde o início do processo, verificou-se uma queda da temperatura dos gases de escape, que é explicada por uma falha na ligação do termopar à placa. Restabelecida a ligação, a temperatura voltou rapidamente a subir, ultrapassando os 60°C cerca de 1 minuto depois e transitando de estado para “Queima”. Este estado manteve-se durante apenas 4 minutos, uma vez que o nível de queima máximo, a que o recuperador se encontrava, tinha programado que o motor de alimentação rodasse ininterruptamente durante o seu período de trabalho, o que fez com que o queimador ficasse com uma grande quantidade de *pellets* para incinerar, aumentando ainda mais a chama e consequentemente, o calor no seu interior.

Ultrapassando os 150°C, que foi a temperatura que ficou estabelecida como sendo a temperatura máxima de funcionamento, o controlador forçou a transição de estado para “Extinção”, desligando o motor de alimentação e colocando os ventiladores ambiente e de gases a rodar ao máximo da velocidade para expulsar o mais depressa possível o calor no interior do aparelho.

Ao longo do teste, a temperatura ambiente chegou aos 26.7°C, enquanto a temperatura da placa, como verificado anteriormente, subiu continuamente, até aos

36.7°C. O estado de “Observação” foi planeado para não ser atingido, dado o curto tempo disponível para a realização do teste, comparado com o tempo necessário para se atingir a temperatura desejada, a não ser que se tivesse considerado um diferencial pequeno.

Durante o processo, foram tiradas algumas fotografias à chama presente no queimador, em função do estado do processo, (Figuras 6.9, 6.10, 6.11 e 6.12) que podem ser visualizadas abaixo.



Figura 6.9 - Chama do queimador na fase de “Ignição”



Figura 6.10 - Chama do queimador no início da fase de “Queima”



Figura 6.11 - Chama do queimador no início da fase de “Extinção”



Figura 6.12 - Câmara de combustão finalizado o processo de aquecimento

Conclui-se que o controlador possui uma máquina de estados funcional e que as interfaces dos atuadores e dos sensores se encontram projetadas corretamente.

Capítulo 7 CONCLUSÕES

No âmbito do trabalho realizado, foi conseguido um considerável avanço em relação ao controlador anteriormente concebido, o que permite aproximá-lo de um protótipo final.

As interfaces dos sensores foram melhoradas, como a do termopar, cujo filtro passa-baixo necessitava de redimensionamento, uma vez que a sua elevada resistência impedia a entrada da ADC de amostrar corretamente o valor de tensão presente à saída do mesmo. (Figuras 3.14 e 3.15). A medição da temperatura dos gases de escape foi dividida em dois troços, em vez de apenas um, tendo em conta a larga gama de temperaturas que se pretendeu medir. Este sensor deve ser cuidadosamente ligado à placa para evitar erros grosseiros na medição da temperatura, como aconteceu na fase de “Ignição” do teste do controlador em ambiente real (Gráfico 6.12). O sensor de temperatura ambiente apenas era utilizado para medir temperaturas entre 10°C e 30°C e neste momento mede entre -3°C e 60°C. A resolução e precisão do processo de medida das temperaturas cumprem os requisitos, apesar de ultrapassarem ligeiramente, para temperaturas relativamente altas medidas pela NTC. A temperatura medida por este sensor segue um cálculo polinomial de ordem cinco, o que acaba por influenciar a resolução em função da temperatura que se pretende medir. Um processo de linearização da sua resposta resolveria, em princípio, esta pequena lacuna.

A temperatura da placa apresenta o inconveniente de aumentar consideravelmente durante os primeiros 20/30 minutos em que se encontra ligado, devido ao aquecimento do regulador de tensão de 5V. Este acontecimento pode levar à diminuição do tempo de operação do próprio *chip*, aconselhando-se a resolução do problema.

A área ocupada pela placa do controlador desenvolvido é inferior relativamente à versão anterior e possui um aspeto mais convidativo e organizado.

A utilização de uma *RTC*, além de permitir um armazenamento da data e hora atual, permitiu o armazenamento permanente de parâmetros, apesar do seu espaço de endereçamento ser reduzido. Num aprimoramento deste controlador, essa limitação deve ser eliminada, pois o número de parâmetros do sistema pode chegar a ser elevado.

A funcionalidade de programação do crono termostato foi implementada com sucesso, apesar de contemplar apenas agendamento diário, com seleção da hora e minuto, única e exclusivamente. A programação semanal pode ser um aspeto perfeitamente exigível por um cliente e deve, por isso, ser tida em conta numa próxima versão.

A funcionalidade preditiva tem a mesma limitação e seguiu uma abordagem simplista do modelo de uma casa, que se justifica nesta fase de desenvolvimento. O seu funcionamento foi validado, junto do engenheiro responsável, com sucesso no decorrer das várias deslocações às instalações da ECOAMA.

A cooperação entre controladores foi igualmente validada, no que diz respeito à comunicação, não tendo sido implementado o seu funcionamento em termos de máquina de estados, visto ser necessário a produção de uma outra placa igual à desenvolvida, além de todo um outro conjunto de sensores e atuadores. Assim, apenas se verificou a ligação de dois microcontroladores semelhantes à placa correndo o mesmo programa, visualizando-se o resultado na aplicação *PC*.

De salientar que, neste trabalho, o controlo da combustão não é muito eficaz (malha aberta), baseando-se apenas na temperatura dos gases de escape, sem ter em conta o estado da chama, da pressão do ar nem da concentração de oxigénio. Apesar de incrementar o seu custo total, são informações que o controlador deve possuir para aumentar o desempenho e eficiência do sistema.

Relativamente à interface com o utilizador, a sua conceção foi criada tendo em conta a opinião e preferências da ECOAMA, resultando em vinte e dois ecrãs disponíveis para fácil navegação por parte do utilizador e do próprio fabricante.

Como acrescento a esta interface, foi desenvolvida uma aplicação *PC* em *JAVA* que pudesse substituir o funcionamento do *LCD*, em caso de avaria, e ajudar na monitorização do processo de aquecimento. O seu aspeto estético não foi a maior preocupação, por isso justifica-se, em abordagens futuras, o seu aprimoramento.

7.1 TRABALHO FUTURO

Em suma, são vários os aspetos que podem ser melhorados e acrescentados, valorizando o controlador desenvolvido:

- **Redução das dimensões da placa do controlador (compactação)**

A utilização de componentes SMD principalmente nos circuitos de baixa potência, assim como um transformador de menores dimensões ajudarão de forma determinante a cumprir esta melhoria.

- **Generalização do controlador para caldeiras e queimadores, além do recuperador de calor**

No âmbito desta dissertação, apenas foi desenvolvido e testado o controlador para funcionar sobre um recuperador de calor. Apesar das poucas diferenças ao nível do funcionamento, este deve poder ser utilizado nos restantes aparelhos identificados.

- **Alargamento do espaço de endereçamento para armazenamento de parâmetros (EEPROM externa ou Flash RAM do PIC32)**

O número de parâmetros deste tipo de sistemas poderá ser elevado e diferir entre caldeiras, recuperadores de calor e queimadores, o que a certa altura ultrapassará, a memória disponível na RTC.

- **Teste e implementação do circuito de acondicionamento para sensor de chama**

A entrada do controlador desenvolvido para sensor de chama não foi projetada, mas em termos de eficiência será um avanço relevante para monitorização da combustão.

- **Introdução de uma interface para sensor de oxigénio (lambda)**

A redução de emissões, assim como a otimização da combustão serão melhorados consideravelmente com a introdução deste sensor.

- **Resolução do problema do considerável aumento da temperatura da placa do controlador durante o tempo em que se encontra ligado**

Uma vez que à entrada do regulador de tensão de 5V se encontra uma diferença de potencial relativamente elevada, será necessário utilizar um transformador com uma relação de transformação superior, utilizar uma resistência para dissipação de calor a montante ou um dissipador de maiores dimensões.

- **Acrescento de programação semanal do processo de aquecimento relativa ao crono termostato e ao preditor**

Certas pessoas preferem definir no início da semana o agendamento do processo de aquecimento para os restantes dias. Nestas circunstâncias, será necessário incluir um campo para seleção do dia da semana.

- **Melhoramento estético e acrescento de funcionalidades da aplicação PC**

Nem todas as funcionalidades desenvolvidas podem ser, neste momento, testadas na aplicação PC, além de apenas funcionar em recuperadores de calor.

- **Elaboração de um modelo mais complexo e robusto de uma casa**

No que diz respeito à predição, pode interessar a elaboração de um modelo de uma casa (ou apenas de um compartimento) mais complexo, onde entrem as principais perturbações a ele associadas, assim como os fatores que contribuem de forma mais determinante para o estabelecimento da temperatura interior.

Ao longo do dia e de dia para dia a temperatura de um compartimento sofre ligeiras alterações, descendo e subindo repetidamente. Isto deve-se ao balanço térmico proporcionado pelos fluxos de ar que sobre ele circulam: o fluxo de calor que entra no compartimento e o fluxo de calor que sai do mesmo. A temperatura exterior tem, assim, uma grande influência.

Neste trabalho deverão ser identificados os modelos do aquecedor e da casa/compartimento.

A Figura 7.1 descreve um exemplo de um diagrama de blocos de um sistema de controlo da temperatura interior de uma casa.

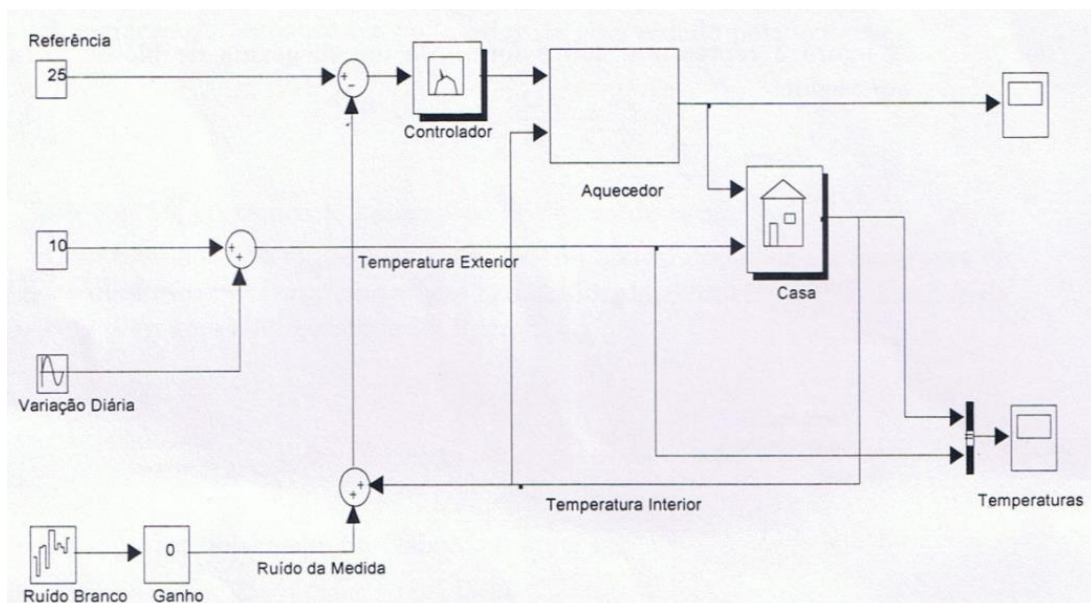


Figura 7.1 – Exemplo de um SC da temperatura interior de uma casa [62]

REFERÊNCIAS

1. ECOAMA. *Site da Empresa [Internet]*. 2013; Available from: <http://www.ecoama.pt/portugues.html>.
2. European Biomass Association. *A Pellet Road Map for Europe [Internet]*. 2008; Available from: http://www.aebiom.org/IMG/pdf/Pellet_Roadmap_final.pdf.
3. Comission of the European Communities. *Biomass action plan [Internet]*. 2005; Available from: http://ec.europa.eu/energy/res/biomass_action_plan/doc/2005_12_07_comm_biomass_action_plan_en.pdf.
4. Edenhofer, O., et al. *Renewable Energy Sources and Climate Change Mitigation - Special Report of the Intergovernmental Panel on Climate Change [Internet]*. 2012; Available from: http://srren.ipcc-wg3.de/report/IPCC_SRREN_Full_Report.pdf.
5. Trømborga, E., et al., *Economic sustainability for wood pellets production – A comparative study between Finland, Germany, Norway, Sweden and the US*. Biomass and Bioenergy, 2013. **57**: p. 68-77.
6. Kang, S., et al., *Development of an air fuel control system for a domestic wood pellet boiler*. Journal of Mechanical Science and Technology, 2013. **27**(6): p. 1701-1706.
7. Oliveira, C. *Futuro da Biomassa no Brasil - Potencial de Biomassa Florestal e Industrial e o Desenvolvimento de Projetos Industriais Sustentáveis [Internet]*. 2014; Available from: http://www.porthuseventos.com.br/congresso_biomassa/Celso_Oliveira.pdf.
8. Alakangas, E. and P. Paju. *Wood Pellets in Finland - technology, economy and market [Internet]*. 2002; Available from: http://www.ateap.cz/new/Pelety_Finsko.pdf.
9. Curran Renewable Energy. *Premium Wood Pellet Fuel [Internet]*. 2014; Available from: <http://www.curranpellets.com/documents/future-of-american-energy.pdf#!premium-wood-pellets/cisy>.
10. Massachusetts Division of Energy Resources. *Wood Pellet Heating - A Reference on Wood Pellet Fuels & Technology for Small Commercial & Institutional Systems (Guidebook) [Internet]*. 2007; Available from: http://www.biomasscenter.org/pdfs/DOER_Pellet_Guidebook.pdf.
11. Plumbset UK LDT. *Carbon Neutral Wood Pellets [Internet]*. 2014; Available from: <http://www.plumbset.co.uk/biomass-boilers.php>.
12. Wood Pellet Line. *Rotary Drum dryer [Internet]*. Available from: <http://www.woodpelletline.com/Rotary-Drum-Dryers.htm>.

13. Philmec. *Flatbed* [Internet]. 2002; Available from: <http://www.philmech.gov.ph/default.asp?page=photos&hd=&mcols=7&Path=images\photos\Postharvest%20Facilities/FLATBED>.
14. PelHeat. *The Beginners Guide To Pellet Production* [Internet]. 2013; Available from: http://www.weedcenter.org/mrwc/docs/The_Beginners_Guide_To_Making_Pellets.pdf.
15. Pellet Process. *Manufacturing Process* [Internet]. 2014; Available from: http://www.pelletprocess.de/?page_id=18&lang=en.
16. Pieter D. Kofman. *The production of wood pellets* [Internet]. 2007; Available from: <http://www.woodenergy.ie/media/woodenergy/content/faq/The%20production%20of%20wood%20pellets.pdf>.
17. Ciolkosz, D. *Renewable and Alternative Energy Fact Sheet* [Internet]. 2009.
18. Audigane, N., et al. *European Pellet Report*. 2012; Available from: <http://www.enplus-pellets.eu/wp-content/uploads/2013/11/D2.2bis-European-pellet-report.pdf>.
19. Fiedler, F., *The state of the art of small-scale pellet-based heating systems and relevant regulations in Sweden, Austria and Germany*. Renewable and Sustainable Energy Reviews, 2004. **8**(3): p. 201-221.
20. Alakangas, E. *New European Pellet Standards* [Internet]. 2010; Available from: http://www.wsed.at/fileadmin/redakteure/WSED/2010/download_presentations/Alakangas_paper_neu.pdf.
21. RIK A Pellets Stoves. *RIKA Brochure* [Internet]. 2014; Available from: <http://www.rika.at/en/katalog-bestellen/>.
22. Salar Lareiras. *Recuperador Calor Damasco NC Bio Bronpi* [Internet]. Available from: <http://www.salar-lareiras.pt/catalogo-produtos/bio-bronpi/recuperadores-de-calor-pellets/biobr-damasco-nc-recuperador-calor-damasco-nc-bio-bronpi.html>.
23. Wood Pecker Energy. *Wood Pellet Boiler Systems* [Internet]. 2010; Available from: <http://rftr.co.uk/downloads/Woodpecker-Brochure.pdf>.
24. Electronics, A. *Fumis LevelTronic - Biomass Level Sensor (Info Sheet)* [Internet]. 2011; Available from: http://www.fumis.si/catalog-documents/INFO_Sheet_Fumis_LevelTronic_eng_WEB.pdf.
25. Obernberger, I. *State-of-the-art of small-scale biomass combustion in boilers* [Internet]. 2008; Available from: http://www.ieabcc.nl/workshops/task32_Dublin_SSC/05%20Obernberger.pdf.
26. Dias, J., M. Costa, and J.L.T. Azevedo, *Test of a small domestic boiler using different pellets*. Biomass and Bioenergy, 2004. **27**(6): p. 531-539.

27. Windhager. *BioWIN 2 -The Pellet Boiler [Internet]*. 2014; Available from: http://www.windhager.co.uk/files/4814/0630/0916/BioWIN2_Brochure_01.07.14.pdf.
28. Fiedler, F., *Combined Solar and Pellet Heating Systems - Study of Energy Use and CO-Emissions*. Malardalen University 2006: p. 1-98.
29. Kedel. *A fully-automated, self-cleaning pellet boiler [Internet]*. 2012; Available from: <http://www.revisionheat.com/pdfs/kedel-brochure.pdf>.
30. Cooke, R. and A. Russell. *Procurement Guidelines for Biomass Heating [Internet]*. 2007; Available from: <http://www.mercia-energy.co.uk/Downloads/BiomassGuidelines07.pdf>.
31. TEM. *Biomass Controllers [Internet]*. Available from: http://www.tem.ch/Bilder/Service_Dokumente/TEM_Biomass_eng.pdf.
32. Tech. *Microprocessor Temperature Regulators [Internet]*. 2009; Available from: <http://santehnika.lv/imgs/files/Tech%20katalogas%5Ben%5D.pdf>.
33. Naturela Electronics. *NPBC-V5 - A high class pellet boiler and burner controller [Internet]*. 2014; Available from: <http://www.naturela-bg.com/en/product/47-0-npbc-v5-universalen-kontroler-za-upravljenie-na-visok-klas-peletni-gorelki-i-kotli/>.
34. ATech Electronics. *Fumis ALPHA - Superior Combustion Control Technology (Info Sheet) [Internet]*. Available from: http://www.fumis.si/catalog-documents/INFO_Sheet_Fumis_ALPHA_eng_WEB.pdf.
35. Moura, N., *Controlador e Device Drivers para um Recuperador de Calor a Pellets (Tese de Mestrado)*. Universidade de Aveiro, 2013.
36. Olimex Ltd. *PIC32-Maxi-Web: User`s Manual [Internet]*. 2013; Available from: <https://www.olimex.com/Products/PIC/Development/PIC32-MAXI-WEB/resources/PIC32-MAXI-WEB.pdf>.
37. Mutambara, A.G.O., *Design and Analysis of Control Systems*1999: Library of Congress
38. Bishop, R.H., *Mechatronic Systems, Sensors, and Actuators: Fundamentals and Modeling*, ed. C. Press2007.
39. Janocha, H., *Actuators: Basics and Applications*2004: Springer Science & Business Media.
40. Probes, S. *Probes for Temperature Measurement - Catalog [Internet]*. 2014; Available from: http://sacet-probes.com/cms/images/immagini_sacet/DOWNLOAD/CATALOG.pdf.
41. Thermometrics. *Type K Thermocouple [Internet]*. 2013; Available from: <http://www.thermometricscorp.com/thermocouple.html>.

42. Texas Instruments. *LM135/LM235/LM335, LM135A/LM235A/LM335A Precision Temperature Sensors (Datasheet)*.
43. Delta Electronics Online Shop. *LM335 Temperature Sensor [Internet]*. 2014; Available from: http://de-onlineshop.nl/index.php?route=product/product&product_id=778.
44. Rauschert. *High-temperature heating elements [Internet]*. 2012; Available from: http://www.rauschert.de/cms/upload/pdf/sonstige_Dokumente/Produktbroschre_HTH_GB.pdf.
45. BCZ Elettromeccanica S.N.C. *Modello 509 [Internet]*. 2005; Available from: <http://bcz.it/>.
46. AACO Manufacturing. *Fume Extrators for Pellet Stoves [Internet]*. 2005; Available from: http://www.aaco.it/j/materiale/prodotti/estrattori/AACO_ST20-73.pdf.
47. AACO Manufacturing. *Fume Extractors and Ventilators for Hot Air Generators [Internet]*. 2005; Available from: http://www.aaco.it/j/materiale/prodotti/ventilatori/AACO_centrifugal.pdf.
48. Maxim Integrated. *DS1307 (Datasheet)*. 2008.
49. Surplustronics shop. *DS1307 Real Time Clock RTC [Internet]*. Available from: <http://www.surplustronics.co.nz/products/1786-ds1307-real-time-clock-rtc>.
50. Redfish Sensors. *NTC Thermistor - Resistance-Temperature Characteristics [Internet]*. Available from: <http://www.redfishsensors.com/PDFs/R805-103F-3435-C.pdf>.
51. Texas Instruments. *MOC3020 through MOC3023 Optocouplers/Optoisolators (Datasheet)*. 1998.
52. Olimex Ltd. *PIC32-T795 development board User's Manual [Internet]*. 2012; Available from: https://www.olimex.com/Products/Duino/PIC32/PIC32-T795/resources/PIC32-T795_um.pdf.
53. Digi-key Corporation Online Shop. *LCD Display NHD-3.5-320240MF-ATXL#-1 [Internet]*. Available from: <http://www.digkey.com/product-detail/en/NHD-3.5-320240MF-ATXL%23-1/NHD-3.5-320240MF-ATXL%23-1-ND/2165878>.
54. Melo, A.P., *Teoria dos Sistema de Controlo Lineares* 2010: Universidade de Aveiro.
55. Wagner, F., et al., *Modeling Software with Finite State Machines - A Pratical Approach* 2006: Auerbach.
56. Feng, T., et al., *System Design, Modeling, and Simulation using Ptolemy II - Chapter 6* 2014: Ptolemy.
57. Pedreiras, P. *StateCharts - Especificação, Modelação e Projecto de Sistemas Embutidos (Universidade de Aveiro)*. 2009; Available from:

<http://ppedreiras.av.it.pt/resources/empse0910/praticas/empse-statecharts-handson.pdf>.

58. Stringhman, G., *Hardware/Firmware Interface Design - Best Practices for Improving Embedded System Development* 2010: Elsevier.
59. Industries, M. *Type K Thermocouple Calibration - Efficient voltage to temperature conversion and temperature measurement using rational polynomial functions [Internet]*. Available from: <http://www.mosaic-industries.com/embedded-systems/microcontroller-projects/temperature-measurement/thermocouple/type-k-calibration-table>.
60. National Institute of Standards and Technology (NIST) *ITS-90 Table for type K thermocouple [Internet]*. 1999; Available from: http://srdata.nist.gov/its90/download/type_k.tab.
61. Fonseca, P., *Sistema de Instrumentação Eletrónica* 2011: Universidade de Aveiro.
62. Mota, A., *Modelos de Sistemas - Versão Beta 0.1*. Departamento de Electrónica Telecomunicações e Informática - Universidade de Aveiro, 2013.

ANEXOS

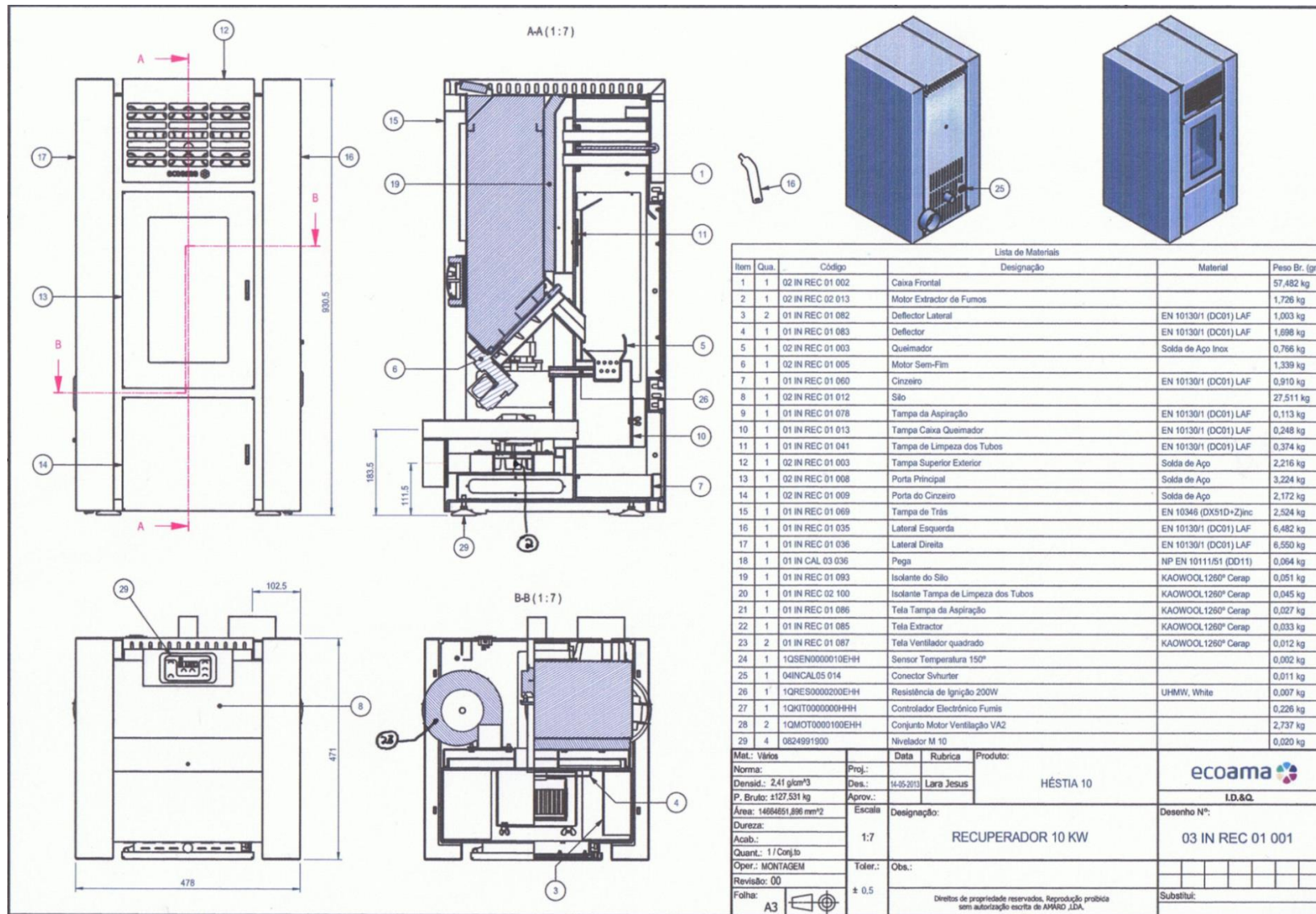


Figura 8.1 - Esquema do recuperador de calor Héstia (ECOAMA)

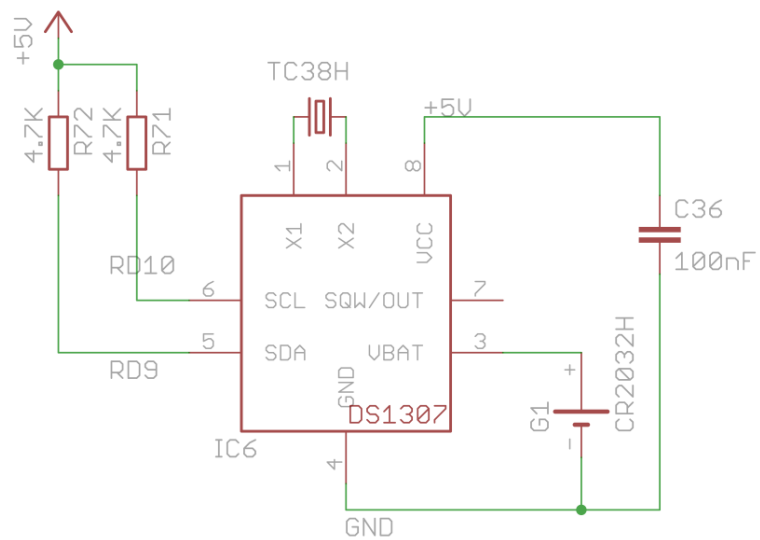


Figura 8.2 - Esquema de ligação da RTC ao microcontrolador

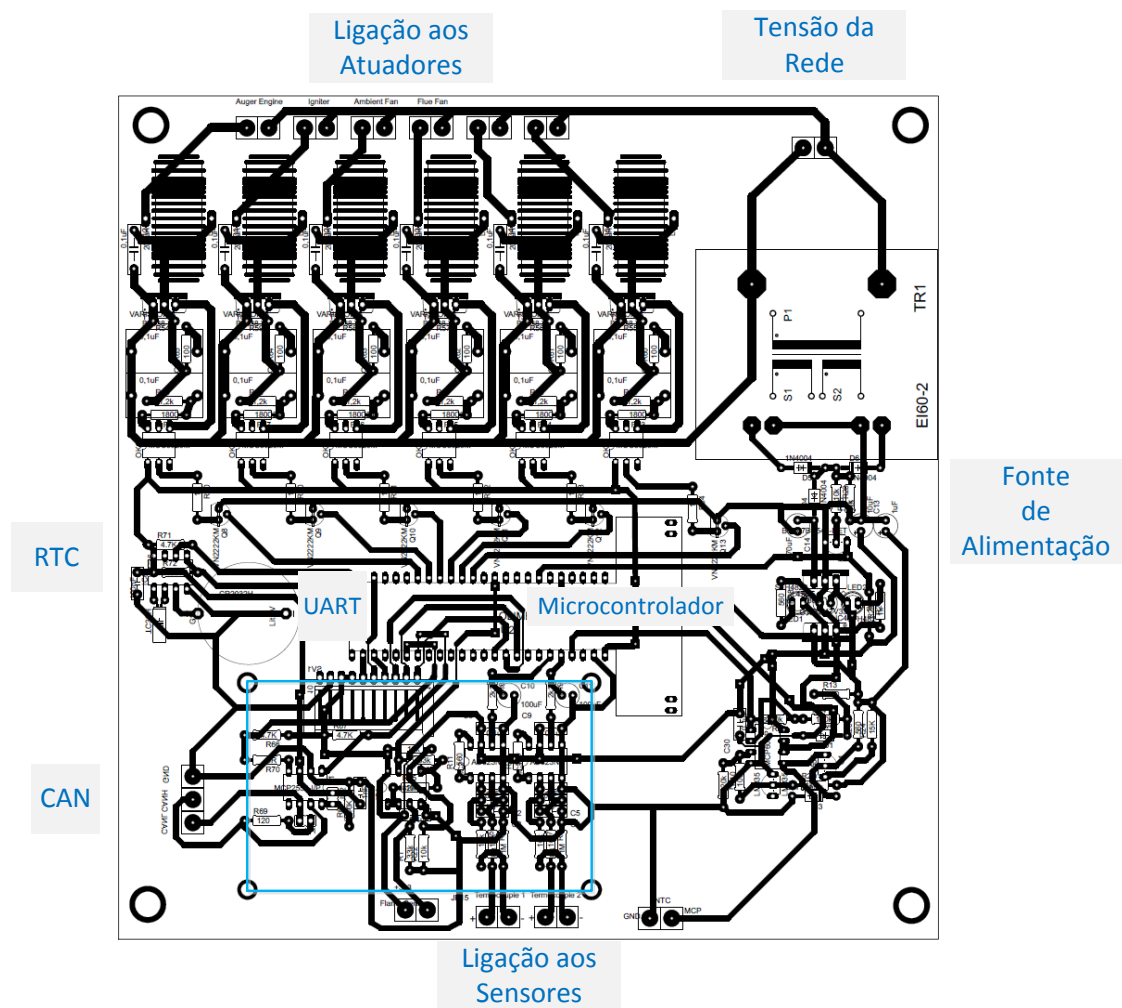


Figura 8.3 - Desenho da PCI do controlador

Endereço	Valor por defeito	Identificação do Parâmetro	Tipo de Parâmetro
0x8	10	Período (s) - Nível 1	Motor de alimentação
0x9	10	Período (s) - Nível 2	
0xA	10	Período (s) - Nível 3	
0xB	10	Período (s) - Nível 4	
0xC	10	Período (s) - Nível 5	
0xD	10	Tempo ligado (%) - Nível 1	
0xE	30	Tempo ligado (%) - Nível 2	
0xF	50	Tempo ligado (%) - Nível 3	
0x10	75	Tempo ligado (%) - Nível 4	
0x11	100	Tempo ligado (%) - Nível 5	
0x12	60	Tempo ligado I (s) - Ignição	
0x13	20	Período (s) – Ignição	
0x14	10	Tempo ligado II (%) - Ignição	
0x15	33	Tensão (V) – Nível 1	Ventilador ambiente
0x16	77	Tensão (V) – Nível 2	
0x17	121	Tensão (V) – Nível 3	
0x18	169	Tensão (V) – Nível 4	
0x19	220	Tensão (V) – Nível 5	
0x1A	33	Tensão (V) – Nível 1	Ventilador dos gases de escape
0x1B	77	Tensão (V) – Nível 2	
0x1C	121	Tensão (V) – Nível 3	
0x1D	169	Tensão (V) – Nível 4	
0x1E	220	Tensão (V) – Nível 5	
0x1F	77	Tensão (V) - Ignição	Temperaturas
0x20	160	Temperatura máxima para os gases de escape (°C)	
0x21	60	Temperatura de transição ignição – Queima (°C)	
0x22	50	Temperatura de transição extinção – Inativa (°C)	
0x23	25	Temperatura desejada (°C)	
0x24	10	Limite máximo para temperatura desejada (°C)	
0x25	35	Limite mínimo para temperatura desejada (°C)	
0x26	0	Hora de início (hh) – Programa 1	Crono termostato
0x27	0	Minuto de Início (mm) – Programa 1	
0x28	0	Hora de fim (hh) – Programa 1	
0x29	0	Minuto de fim (mm) – Programa 1	
0x2A	15	Temperatura desejada (°C) – Programa 1	
0x2B	0	Hora de início (hh) – Programa 2	
0x2C	0	Minuto de Início (mm) – Programa 2	
0x2D	0	Hora de fim (hh) – Programa 2	
0x2E	0	Minuto de fim (mm) – Programa 2	
0x2F	20	Temperatura desejada (°C) – Programa 2	
0x30	0	Hora de início (hh) – Programa 3	
0x31	0	Minuto de Início (mm) – Programa 3	
0x32	0	Hora de fim (hh) – Programa 3	
0x33	0	Minuto de fim (mm) – Programa 3	
0x34	25	Temperatura desejada (°C) – Programa 3	
0x35	0	Identificador do programa ativo / Identificador do estado do crono termostato	
0x36	1	Número Identificador	

0x37	40	Temperatura de entrada em funcionamento (°C)	Controladores
0x38	3	Nível de queima	
0x39	4	Nível máximo de queima	-
0x3A	123	Palavra-Passe	-
0x3B	0	Temperatura inicial (°C)	Calibração
0x3C	0	Temperatura final (°C)	
0x3D	0	Tempo de calibração (min)	
0x3E			
0x3F			

Tabela 8.1 - Mapeamento dos parâmetros do sistema na memória da RTC

Origem	Comando	Ação despoletada
Botão "Set"	"fd"	O vetor com os parâmetros do motor de alimentação é atualizado
	"af"	O vetor com os parâmetros do ventilador ambiente é atualizado
	"ff"	O vetor com os parâmetros do ventilador de escape é atualizado
	"tp"	O vetor com as temperaturas parametrizadas é atualizado
	"idgt"	Atualiza o valor do ID, temperatura e nível de queima de funcionamento do controlador (<i>gateway</i>)
	"idc1t"	Atualiza o valor do ID, temperatura e nível de queima de funcionamento do controlador 1 a ele ligado
	"idc2t"	Atualiza o valor do ID, temperatura e nível de queima de funcionamento do controlador 2 a ele ligado
	"idc3t"	Atualiza o valor do ID, temperatura e nível de queima de funcionamento do controlador 3 a ele ligado
	"idc4t"	Atualiza o valor do ID, temperatura e nível de queima de funcionamento do controlador 4 a ele ligado
Botão "Set default"	"fdd"	O vetor com os parâmetros do motor de alimentação é atualizado e RTC guarda a informação
	"afd"	O vetor com os parâmetros do ventilador ambiente é atualizado e RTC guarda a informação
	"ffd"	O vetor com os parâmetros do ventilador de escape é atualizado e RTC guarda a informação
	"tpd"	O vetor com as temperaturas parametrizadas é atualizado e RTC guarda a informação

Tabela 8.2 - Comandos de atualização de parâmetros

Origem	Comando	Ação despoletada
Botão "ON"	"f1"	Ligar motor de alimentação
	"i1"	Ligar resistência de ignição
	"a1"	Ligar ventilador ambiente
	"F1"	Ligar ventilador de escape
Botão "OFF"	"f0"	Desligar motor de alimentação
	"i0"	Desligar resistência de ignição
	"a0"	Desligar ventilador ambiente
	"F0"	Desligar ventilador de escape
Botão "-"	"a-"	Diminuir tensão aplicada ao ventilador ambiente
	"F-"	Diminuir tensão aplicada ao ventilador de escape
Botão "+"	"a+"	Aumentar tensão aplicada ao ventilador ambiente
	"F+"	Aumentar tensão aplicada ao ventilador de escape

Tabela 8.3 - Comandos utilizados no teste dos atuadores

Origem	Comando	Ação despoletada
Caixas de Seleção	"p1"	Programa 1 agendado
	"p2"	Programa 2 agendado
	"p3"	Programa 2 agendado
	"p0"	Nenhum programa é agendado
Botão "Save"	"sv1"	Desligar Motor de alimentação
	"sv2"	Desligar resistência de Ignição
	"sv3"	Desligar ventilador ambiente
	"sv4"	Desligar ventilador de escape

Tabela 8.4 - Comandos de agendamento do crono termostato

Origem	Comando	Ação despoletada
Botão "START"	"str"	Dar início ao processo
Botão "STOP"	"stp"	Terminar o processo

Tabela 8.5 - Comandos de solicitação de início e fim de processo

8.1 Ecrãs desenhados para o LCD

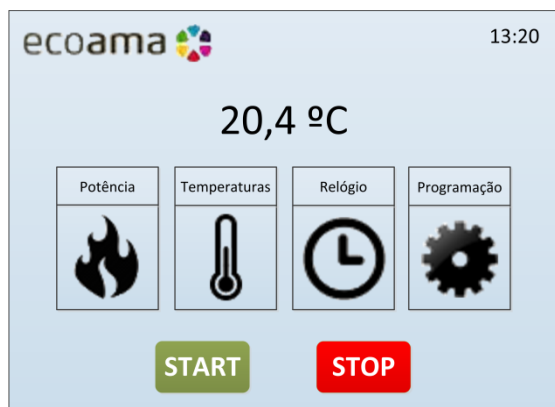


Figura 8.4 - Processo inativo

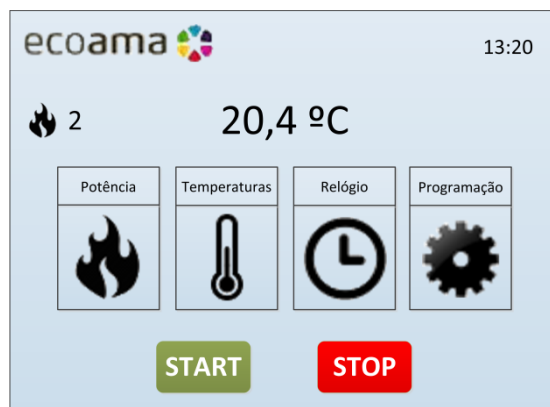


Figura 8.5 - Processo ativo

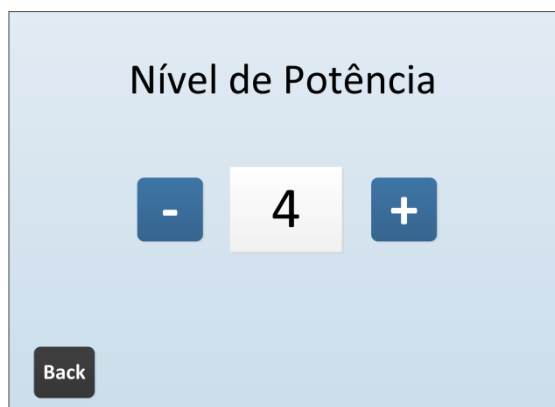


Figura 8.6 - Seleção de nível máximo de queima



Figura 8.7 - Menu de temperaturas

Temperatura Ambiente 21,4 °C

Temperatura da Placa 25,2 °C

Temperatura dos Gases de Escape 65,6 °C

Back

Figura 8.8 - Mostrador de temperaturas

Temperatura Desejada

- 25 °C +

Back

Figura 8.9 - Seleção de temperatura desejada

Relógio

Alterar Data e Hora atuais

Crono Termostato

OFF

Preditor

Back

Figura 8.10 - Menu de relógio

15:30

5 Agosto 2014 - Terça-Feira

Ano - 2014 +

Mês - 8 +

Dia - 5 +

Hora - 20 +

Minuto - 10 +

OK

Back

Figura 8.11 - Atualização da data e hora atuais

15:30

< P1 >

Início Fim Temp

18:30 - + 19:30 - + 26 +

Back

OFF

Figura 8.12 - Programação do crono termostato

15:30

Preditor

HH:MM Temp

18:30 - + 26 +

Back

OFF

Figura 8.13 - Predição

Senha

1 2 3

4 5 7

6 8 9

0 ,

<- ENTER

Back

Figura 8.14 - Inserção de palavra-passe

Menu de Programação

Programar Parâmetros

Testar Atuadores

Bloquear Teclado

Calibrar

Back

Figura 8.15 - Menu de programação

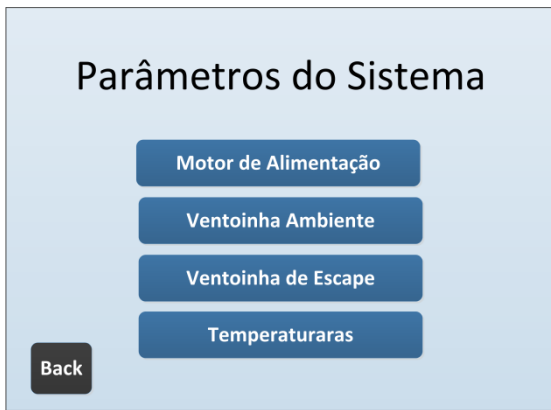


Figura 8.16 - Menu de parâmetros

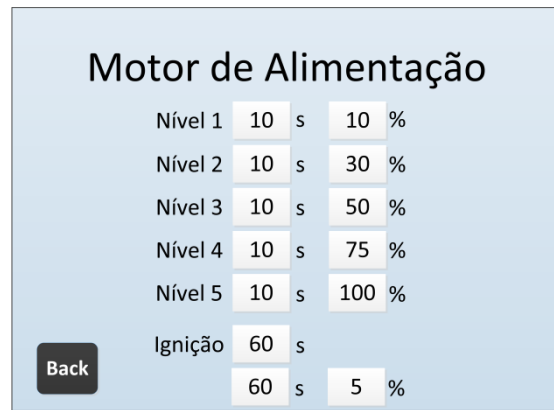


Figura 8.17 - Config. parâmetros do MA

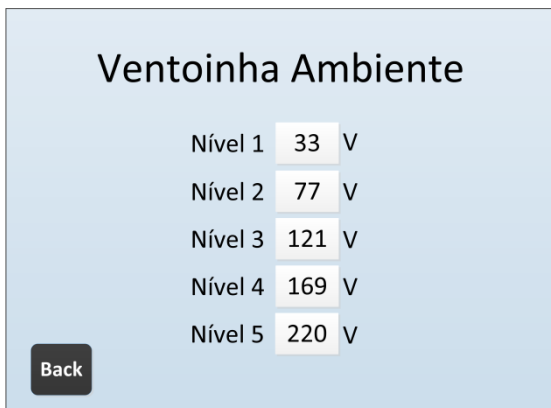


Figura 8.18 - Config. parâmetros do VA

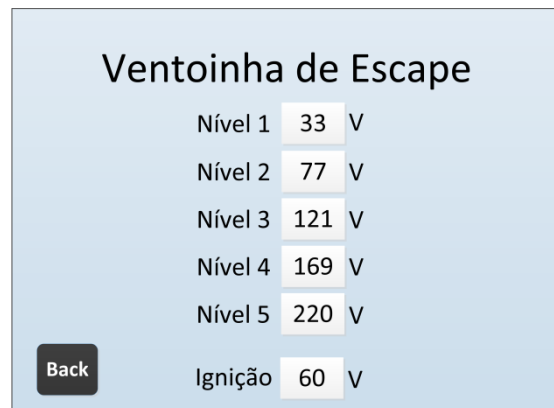


Figura 8.19 - Config. parâmetros do VE



Figura 8.20 - Config. temperaturas

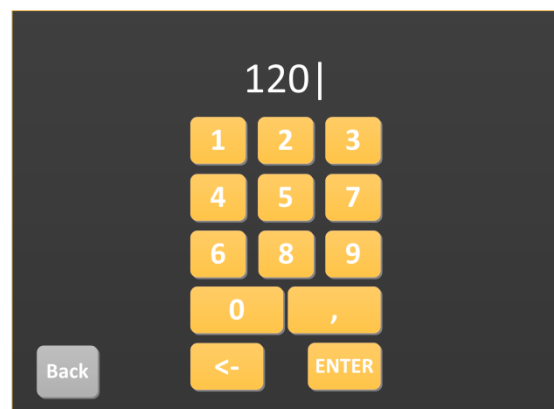


Figura 8.21 - Teclado virtual



Figura 8.22 - Teste aos atuadores

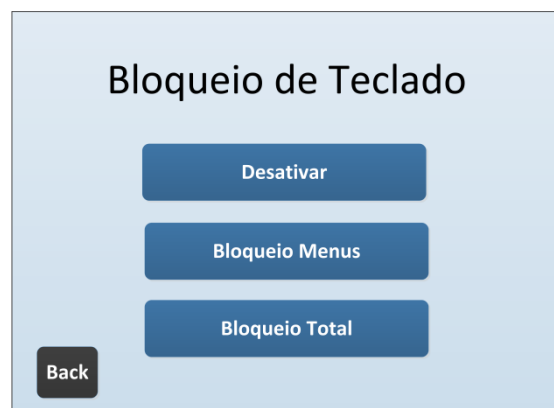


Figura 8.23 - Menu de bloqueio do teclado



Figura 8.24 - Calibração

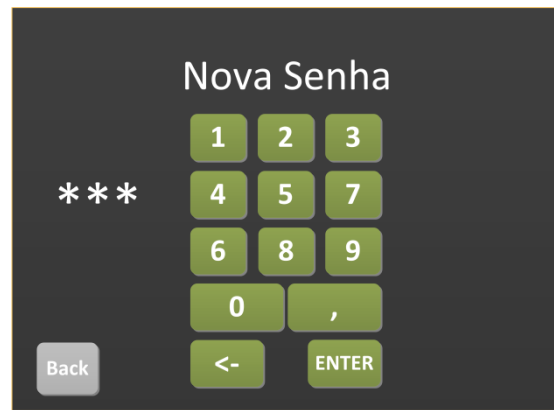


Figura 8.25 - Inserção de nova palavra-passe

8.2 Interface Inter Integrated Circuit (I2C)

Esta interface foi desenvolvida pela Philips Semiconductor em 1992, lançando a sua última versão em 2007. Corresponde a um tipo de comunicação série bidirecional, *half-duplex*, orientado ao *byte* com transações *Master-Slave*. Numa escrita, um *Master* transmissor envia dados para um *Slave* recetor, enquanto na leitura um *Master* recetor lê dados de um *Slave* transmissor.

Sinalização

O barramento é composto por apenas 2 fios: um deles para a linha de dados (*SDA*) e outra para a linha de relógio (*SCL*). Em cada período do relógio é transferido um *bit* de dados, sendo que estes apenas ficam válidos quando *SCL* = "1" e apenas podem ser alterados quando *SCL* = "0". Contudo, existem exceções nas condições, que podem ser de dois tipos: *start* e *stop*. O primeiro sinaliza o início de uma transferência de dados e ocorre na transição de "1" para "0" em *SDA*, enquanto a linha *SCL* se encontra no nível lógico "1". A segunda corresponde ao seu término e ocorre na transição de "0" para "1" em *SDA* quando *SCL* = "1". Pode-se, por isto, dizer que o barramento se encontra ocupado entre um *start* e um *stop* e livre entre um *stop* e um *start*.

Transmissão de dados

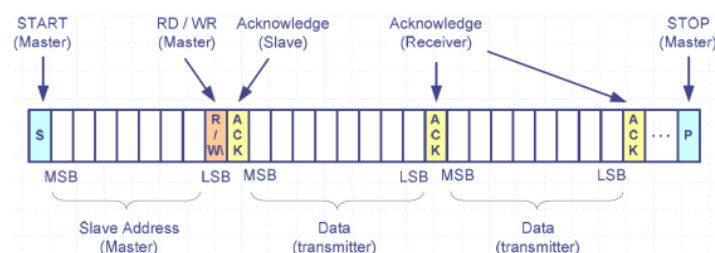


Figura 8.26 - Tramas I2C para transferência de dados

Traduzindo de forma sequencial a Figura 8.26, serão referidos abaixo os passos a seguir para uma correta transmissão de dados:

- 1 – *Master* envia um *Start Bit*, sinalizando o início da transmissão;
- 2 – *Master* envia 7 *bits* de endereçamento do *Slave* com o qual pretende comunicar acrescido do *bit* de operação, leitura ou escrita;
- 3 – Se o endereçamento for efetuado de forma correta, o *Slave* deve responder com um *bit* de confirmação (*Acknowledge*).
- 4 – Se a operação a efetuar for de escrita, então o *Master* começa a transferir os dados, 8 de *bits* por *slot*, esperando sempre o *bit* de confirmação do *Slave*. Quando todos os dados tiverem sido transferidos, então o *Master* envia um *stop bit* e dá por encerrada a transação.

Se o *Master* pretender efetuar uma leitura, o *Slave* começa de imediato a transferir informação para o barramento, com a confirmação a ser dada pelo *Master* a cada 8 *bits*. Quando este tiver recebido todos os dados de que necessita, envia um *bit* de *Not Acknowledge* e de seguida um *stop bit*, libertando de novo o barramento.

Taxas de Transmissão

A taxa de *bits* transferidos por segundo classifica-se em 4 tipos, sendo eles:

- Standard mode: até 100 Kbit/s
- Fast mode: até 400 Kbit/s
- Fast mode plus: até 1 Mbit/s
- High Speed: até a 3,4 Mbit/s

Barramento Multi-Master

Vários *Masters* podem pretender iniciar uma nova transferência de dados no mesmo instante. É, por isso necessário que os seus relógios se mantenham sincronizados e um processo de arbitragem para determinar aquele que vai ter acesso ao barramento. Ambos utilizam a técnica de *bit* dominante (“0”) / recessivo (“1”), ou seja, o *bit* dominante sobrepõe-se sempre ao *bit* recessivo.

Sincronização de Relógio dos Masters

Logo que na linha *SCL* ocorra uma transição “1” -> “0”, todos os *Masters* são forçados a colocar os respetivos relógios a “0” durante um determinado intervalo de tempo determinado por cada um. Decorrido esse tempo para um dos *Masters*, este liberta a linha *SCL*, o que não significa que esta passe automaticamente para o nível lógico “1”, uma vez que outros *Masters* podem ainda não ter terminada a contagem a “0”. Nesta

situação todos os *Masters* que tiverem libertado a linha *SCL* passam para um estado de *wait*, ficando a aguardar que a linha passe a “1”. A partir daqui, os *Masters* iniciam a sua contagem a “1”, contudo, nesta situação basta que o primeiro *Master* termine esse tempo para colocar automaticamente a linha de relógio a “0”. E o processo repete-se.

Arbitragem

Neste processo, os *Masters* que pretendem aceder ao barramento verificam o novo bit transferido e comparam com aquele que eles próprios enviaram. O *Master* que tenha enviado um *bit* recessivo e tendo lido da linha de dados um *bit* dominante, perde automaticamente o processo de arbitragem e:

- Retira-se libertando a linha de dados, comutando para o modo *Slave*;
- Só pode voltar a tentar aceder ao barramento quando a linha de dados voltar ao estado livre;

Exemplos de aplicações

Exemplos de dispositivos que podem comunicar segundo esta interface são:

- Memórias;
- *ADCs* e *DACs*
- *LCD* drivers
- *Real-Time Clocks*
- *TVs* e Monitores
- *CPUs* e *Motherboards*

8.3 Serial Peripheral Interface (SPI)

Definida e desenvolvida pela Motorola, esta interface de comunicação é utilizada num grande número de dispositivos, como por exemplo: sensores, cartões de memória, memórias externas, *ADCs*, *DACs*, *Displays LCD*, máquinas fotográficas, microcontroladores, etc.

É utilizado em ligações a curta distância e caracteriza-se por uma arquitetura Master-Slave com ligação ponto a ponto. Ao contrário do *I2C*, a comunicação é bidirecional *full-duplex*, ou seja, pode dar-se nos dois sentidos ao mesmo tempo. Funciona em modo “*Data-Exchange*”, sendo que cada dispositivo ao fim de *N* ciclos de relógio envia uma palavra de *N bits* para o barramento e recebe outra com o mesmo comprimento.

O sinal de relógio é gerado pelo Master e disponibilizado a todos os *Slaves* – relógio explícito do *Master*. Nesta arquitetura, apenas pode existir um *Master* que inicia e controla a transferência de dados, sendo um único *Slave* é selecionado de cada vez.

Esquema de Princípio

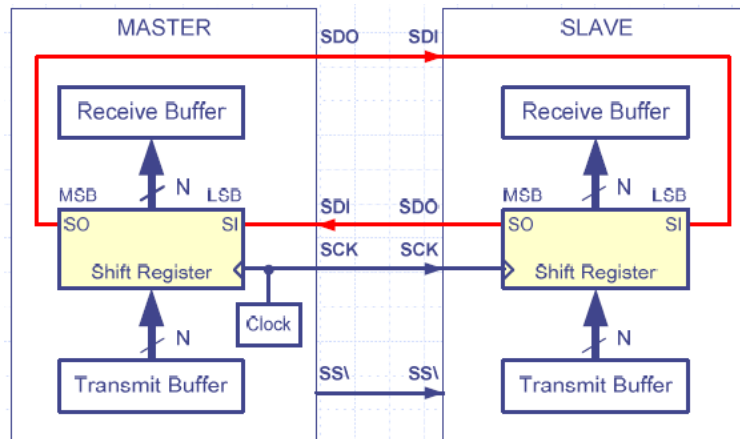


Figura 8.27 - Modelo de comunicação SPI

Em cada dispositivo existe um *shift-register*, ao qual se juntam dois *buffers*: de transmissão e recepção (Figura 8.27).

A comunicação dá-se, então da seguinte forma: a informação é enviada do *buffer* de transmissão para o *shift-register* que, por sua vez a reencaminha para outro *shift-register*, depositando o seu conteúdo no *buffer* de recepção respetivo.

Sinalização

São utilizadas duas linhas para dados, *Serial Data In* (SDI) e *Serial Data Out* (SDO) e outras duas para controlo, *Serial Clock* (SCK) e *Slave Select* (SS\). As linhas de dados são utilizadas para transferência de dados entre os *shift-registers* dos dispositivos.

A linha de relógio (SCK) tem um *duty-cycle* de 50% e é utilizada pelo Master para definir a transição (ascendente ou descendente) em que coloca um bit de informação no barramento. A transição oposta é escolhida pelo *Slave* para o armazenar. A frequência da onda deve ser igual ou inferior à máxima suportada pelo *Slave* com o qual vai comunicar.

A linha de seleção do *Slave* (SS\) como o nome indica é utilizada para determinar aquele com o qual o *Master* comunica a cada instante. Esta linha, quando inativa, pára o sinal de relógio, deixando-o estável num nível lógico definido previamente.

Arquiteturas de Ligação

Existem duas arquiteturas principais de ligação *SPI*. A Uma delas atribui sinais independentes de ativação a cada *Slave* (Figura 8.28), ou seja, apenas um está ativo em cada instante. Todos os outros ficam em alta impedância.

Na segunda arquitetura (Figura 8.29), o sinal de *Slave Select* é comum a vários dispositivos, sendo que a saída de dados de cada um se encontra ligada à entrada de dados do seguinte. Esta “cooperação” de *Slaves* permite transformá-los num *Slave* de maior dimensão.

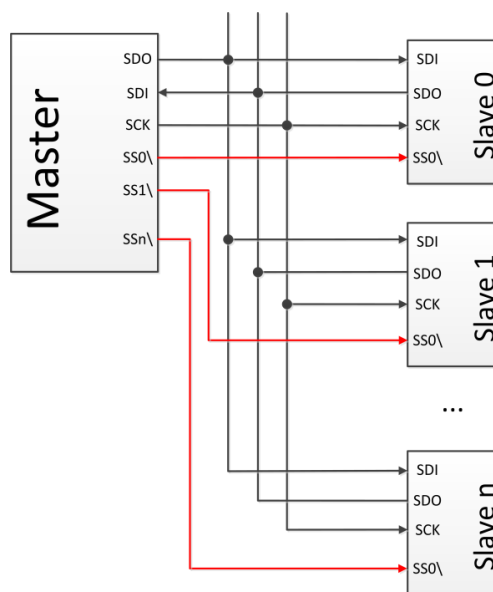


Figura 8.28 - Ligação Independente

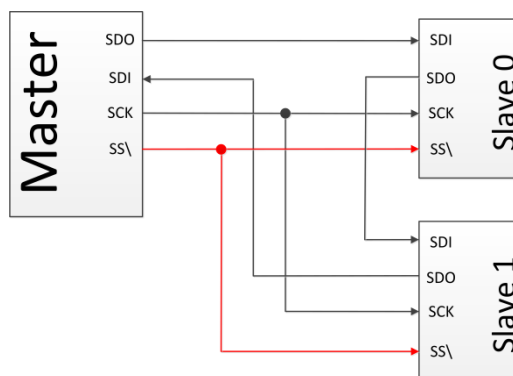


Figura 8.29 - Ligação em cascata

Tipos de Transferências

- Bidirecional – em ambos os sentidos é trocada informação válida ao mesmo tempo

- *Master* → *Slave* (Operação de escrita) – São transferidos dados válidos para o *Slave* e ignorados os que são recebidos na volta.
- *Slave* → *Master* (Operação de leitura) – São recebidos dados válidos do *Slave* e ignorados aqueles que lhe são enviados

8.4 Controller Area Network (CAN)

Esta interface de comunicação foi desenvolvida em 1991 pela *Bosch* e utiliza comunicação diferencial em par entrançado. Pode funcionar com taxas de transmissão até 1 *Mbit/s* e utilizar até 40 nós no mesmo barramento. A capacidade de detecção de diferentes tipos de erros, a tolerância a interferências eletromagnéticas e a baixa probabilidade de não detecção de erros de transmissão fazem com que o *CAN* apresente uma elevada robustez.

Características

O *CAN* é um barramento série *multi-master*, ou seja, qualquer nó do barramento pode produzir e iniciar uma transmissão. Utiliza a relação produtor-consumidor, podendo ainda ocorrer transmissão em *broadcast*. A comunicação é bidirecional, *half-duplex*, com relógio implícito, e a informação produzida é dividida em mensagens (encapsuladas em tramas), cada uma delas possuindo um *ID* único que define a sua prioridade no acesso ao barramento. Quando menor o *ID*, maior será a prioridade já que o processo de arbitragem entre masters utiliza a técnica de *bit* dominante/recessivo.

Topologia de Rede

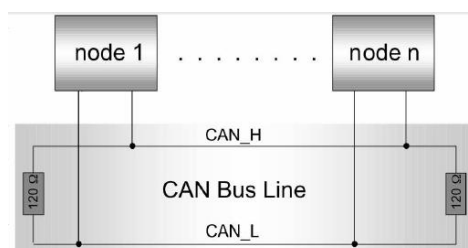


Figura 8.30 - Topologia da rede *CAN*

Um nó liga-se ao barramento através de duas linhas: *CAN_H* e *CAN_L*, necessitando de uma resistência de terminação entre elas na ordem dos 120 Ω (Figura 8.30). Internamente, cada nó necessita de um *transceiver* para conversão de níveis de tensão entre o barramento propriamente dito e o controlador *CAN* interno do microcontrolador/CPU (Figura 8.31). Como dito anteriormente, a comunicação é

diferencial, o que significa que na receção de dados o *transceiver* discrimina o nível lógico (“0” ou “1”) pela diferença entre os dois sinais constituintes do barramento, *CAN_H* e *CAN_L*, cujo resultado é enviado para a linha *Rx* do controlador *CAN*. Na transmissão, o nível lógico em *Tx* é transformado em duas tensões distintas em *CAN_H* e *CAN_L*.

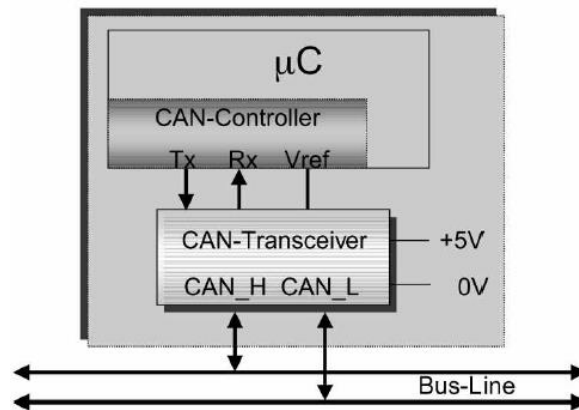


Figura 8.31 - Constituição geral de um nó da rede *CAN*

No que ao controlador *CAN* diz respeito, destacam-se os blocos internos (Figura 8.32):

- Implementação do Protocolo em *hardware*;
- *CPU interface* – assegura a comunicação com o CPU/microcontrolador;
- *Hardware Acceptance Filter* – responsável por filtrar as mensagens recebidas através do seu *ID*;
- *Buffer* de Mensagens para Transmissão – guarda os dados antes de os enviar;
- *Buffer* de Mensagens de Receção – guarda os dados para poderem ser acedidos pelo utilizador.

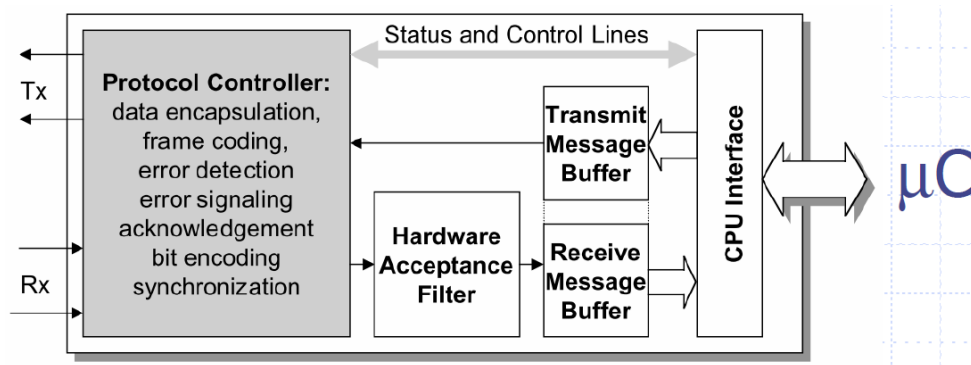


Figura 8.32 - Controlador *CAN*

Tipos de tramas

- Dados – enviada por um nó produtor para um ou mais consumidores;
- Remote Transmission Request – enviada por um nó consumidor a solicitar uma determinada trama de dados;
- Erro – usada para reportar um erro e sobrepõe-se a qualquer transmissão que se esteja a desenrolar;
- Overload – usada para retardar o envio de nova trama, na situação em que não tenha conseguido processar a última trama enviada para o barramento.

Formato da Trama *Standard*

Uma trama *CAN Standard* (Figura 8.33) é composta sete grupos de *bits*. O primeiro é *SOF* (*Start of Frame*) que correspondem a um *bit* dominante (“0”) e é usado para sincronização do relógio dos nós recetores. Segue-se o campo de arbitragem constituído por 11 *bits* para o identificador da mensagem e 1 *bit* (*RTR*) para distinção entre trama de dados de *Remote Transmission Request*. Este campo define o *master* que pode iniciar a transmissão. O campo de controlo é constituído por 6 *bits*, dos quais o primeiro, extensão identificadora, determina se se trata de uma trama *Standard* ou *Extended*. Seguem-se um *bit* dominante reservado, *RB0*, e 4 *bits* que especificam o número de *bytes* de dados da mensagem.

O campo de dados propriamente dito da mensagem tem espaço para 8 *bytes* de dados. Segue-se o campo *CRC*, propositadamente incluído para deteção de erros, com uma sequência de 16 *bits*. Esta sequência é calculada pelo transmissor e recetor da mensagem, com base nos *bits* transmitidos, que caso difiram é detetado um erro. O campo de *Acknowledge* é responsável pela validação da trama. Assim, o produtor envia um *bit* recessivo, que é sobreposto por um ou mais consumidores com um *bit* dominante. Finalmente são introduzidos 7 *bits* recessivos sinalizando o final da trama.

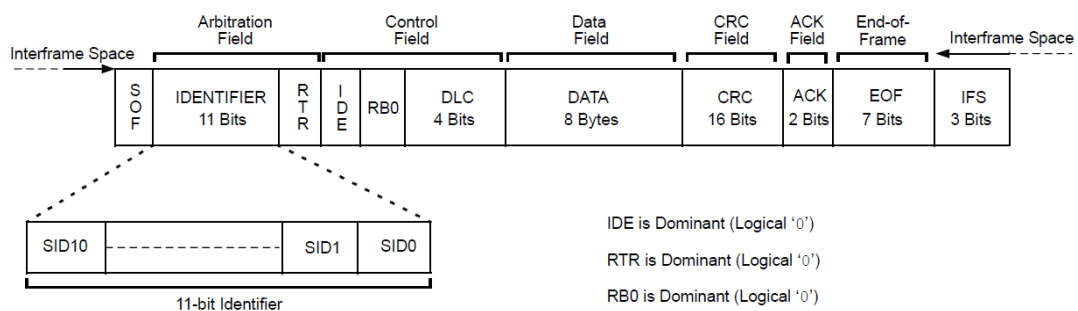


Figura 8.33 - Trama *CAN Standard*

8.5 Pinos do LCD Flex-Cable

Signal	Pin No. (3.5")	Description
LED K	1,2	LED Cathode
LED A	3,4	LED Cathode
DISP	8	General purpose output pin for LCD Display Enable, push-pull output, 4/8mA sink/source current. Control by writing to Bit 7 of REG_GPIO register.
HSYNC	36	LCD Horizontal Sync, push-pull output, 4/8mA sink/source current.
VSYNC	37	LCD Vertical Sync, push-pull output, 4/8mA sink/source current.
DCLK	38	LCD Pixel Clock, push-pull output, 4/8mA sink/source current.
DE	52	LCD Data Enable, push-pull output, 4/8mA sink/source current.
R2	14	Bit 2 of Red RGB signals, push-pull output, 4/8mA sink/source current.
R3	15	Bit 3 of Red RGB signals, push-pull output, 4/8mA sink/source current.
R4	16	Bit 4 of Red RGB signals, push-pull output, 4/8mA sink/source current.
R5	17	Bit 5 of Red RGB signals, push-pull output, 4/8mA sink/source current.
R6	18	Bit 6 of Red RGB signals, push-pull output, 4/8mA sink/source current.
R7	19	Bit 7 of Red RGB signals, push-pull output, 4/8mA sink/source current.
G2	22	Bit 2 of Green RGB signals, push-pull output, 4/8mA sink/source current
G3	23	Bit 3 of Green RGB signals, push-pull output, 4/8mA sink/source current
G4	24	Bit 4 of Green RGB signals, push-pull output, 4/8mA sink/source current
G5	25	Bit 5 of Green RGB signals, push-pull output, 4/8mA sink/source current
G6	26	Bit 6 of Green RGB signals, push-pull output, 4/8mA sink/source current
G7	27	Bit 7 of Green RGB signals, push-pull output, 4/8mA sink/source current
B2	30	Bit 2 of Blue RGB signals, push-pull output, 4/8mA sink/source current.
B3	31	Bit 3 of Blue RGB signals, push-pull output, 4/8mA sink/source current.
B4	32	Bit 4 of Blue RGB signals, push-pull output, 4/8mA sink/source current.
B5	33	Bit 5 of Blue RGB signals, push-pull output, 4/8mA sink/source current.
B6	34	Bit 6 of Blue RGB signals, push-pull output, 4/8mA sink/source current.
B7	35	Bit 7 of Blue RGB signals, push-pull output, 4/8mA sink/source current.
XP	48	X +ve touch
YM	49	Y -ve touch
XM	50	X -ve touch
YP	51	Y +ve touch
3V3	9,10,11,41,42	3V3 Power
GND	12,13,20,21,28,29,53,54	Ground
NC	5,6,7,39,40,43,44,45,46,47	No connect

Tabela 8.6 - Funções de todos os pinos de entrada/saída do LCD